

(19)



JAPANESE PATENT OFFICE

## PATENT ABSTRACTS OF JAPAN

(11) Publication number: **02235146 A**(43) Date of publication of application: **18.09.90**

(51) Int. Cl.

**G06F 9/46**  
**G03G 15/00**  
**H04N 1/00**  
**H04N 1/23**

(21) Application number: **01057280**(22) Date of filing: **08.03.89**(71) Applicant: **FUJI XEROX CO LTD**

(72) Inventor: **OI KOICHI**  
**HOSHI TAKASHI**

**(54) MONITOR CONTROL SYSTEM FOR RECORDING DEVICE**

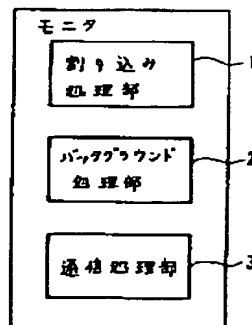
executed, the normal monitor processing is executed without generating any trouble.

**(57) Abstract:**

COPYRIGHT: (C)1990,JPO&amp;Japio

**PURPOSE:** To allow a monitor to execute normal processing based upon an interruption without generating a trouble even when task processing with a long processing time exists by providing the monitor with a background processing part, and at the time of interrupting the processing of the processing part, executing processing based upon the interruption.

**CONSTITUTION:** The calculation processing of an IPS requiring a long processing time, the formation of TRC conversion processing, the switching of a CRT screen, etc., are regarded as background processing with a low priority level. Processing to be surely executed such as timer processing, I/O processing and transmitting/receiving processing is regarded as processing with a high priority level. In each interruption in each prescribed time, the interruption is executed by an interruption processing part 1, and after ending the processing, background processing is executed by a background processing part 2. Since the low level processing is executed by utilizing the idle-time of a CPU in which normal monitor processing is not



BEST AVAILABLE COPY

## ⑫ 公開特許公報(A)

平2-235146

⑤Int. Cl.<sup>5</sup>

G 06 F 9/46  
G 03 G 15/00  
H 04 N 1/00  
1/23

識別記号

3 4 0 B  
1 0 2  
1 0 6 B  
Z

庁内整理番号

8945-5B  
8004-2H  
7334-5C  
6940-5C

④公開 平成2年(1990)9月18日

審査請求 未請求 請求項の数 19 (全46頁)

⑥発明の名称 記録装置におけるモニタ制御方式

⑪特 願 平1-57280

⑫出 願 平1(1989)3月8日

⑬発 明 者 大 井 浩 一 神奈川県海老名市本郷2274番地 富士ゼロックス株式会社  
海老名事業所内  
⑬発 明 者 星 孝 志 神奈川県海老名市本郷2274番地 富士ゼロックス株式会社  
海老名事業所内  
⑭出 願 人 富士ゼロックス株式会 東京都港区赤坂3丁目3番5号  
社  
⑭代 理 人 弁理士 蛭川 昌信 外5名

## 明 細 書

## 1. 発明の名称

記録装置におけるモニタ制御方式

## 2. 特許請求の範囲

(1) 所定間隔毎の割り込みにより処理を実行する割り込み処理部と、バックグラウンド処理部とを有し、バックグラウンド処理部は常時バックグラウンド処理を行うと共に、割り込みにより処理を中断し、バックグラウンド処理中断時に割り込み処理部の通常処理を行うようにしたことを特徴とする記録装置におけるモニタ制御方式。

(2) 所定間隔毎の割り込みは、10 msec割り込みである請求項1記載の記録装置におけるモニタ制御方式。

(3) バックグラウンド処理部は、パワーONにより起動される請求項1または2記載の記録装置におけるモニタ制御方式。

(4) バックグラウンド処理部が行う処理は、予め登録されたタスクである請求項1または2記載の記録装置におけるモニタ制御方式。

(5) 登録されたタスクのステートを、タスクステートテーブルにより管理する請求項4記載の記録装置におけるモニタ制御方式。

(6) 登録された各タスクに対して処理ループ回数をカウントする複数のループカウンタが設定されている請求項4記載の記録装置におけるモニタ制御方式。

(7) 登録されたタスクの処理が終了すると、登録タスク数をディクレメントする請求項4記載の記録装置におけるモニタ制御方式。

(8) 登録されたタスクの処理終了のステートは、所定時間保持するようにした請求項2記載の記録装置におけるモニタ制御方式。

(9) バックグラウンド処理は、10 msecを越える演算処理である請求項1記載の記録装置におけるモニタ制御方式。

(10) バックグラウンド処理部が行うタスクの登録、抹消はスーパーバイザーコールにより行う請求項1記載の記録装置におけるモニタ制御方式。

(11) さらに、通信速度の異なる複数の通信処理部を備えた請求項1記載のモニタ制御方式。

(12) 通信処理部の1つは高速通信処理部であり、データが空になったことによる通信割り込みと、10 msec割り込みにより送信処理が起動される請求項11記載のモニタ制御方式。

(13) データが空になったことによる通信割り込みと、10 msec割り込みが同時に発生したときは、一方の割り込みをマスクする請求項12記載のモニタ制御方式。

(14) 高速通信処理部は、送受信データバッファをそれぞれ複数個有し、データの書き込み/読み取りと、データの送受信とを同時に行えるようにした請求項12記載のモニタ制御方式。

(15) 高速通信処理部は、パケット単位でデータの送受信を行う請求項12記載のモニタ制御方式。

(16) 高速通信処理部は、送受信データバッファを通信相手に割り振っておくことを特徴とする請求項14記載のモニタ制御方式。

(17) 通信処理部の1つがポーリング方式通信処理部であり、バイト単位でデータの送受信を行う請求項11記載のモニタ制御方式。

(18) ポーリング方式通信処理部は、データの授受があったときに発生する割り込みで送受信処理が起動される請求項17記載のモニタ制御方式。

(19) ポーリング方式通信処理部は、回線エラー監視処理部と回線エラーをアプリケーションに知らせる回線エラー検知処理部とを有する請求項17記載のモニタ制御方式。

### 3. 発明の詳細な説明

#### 〔産業上の利用分野〕

本発明は記録装置の制御方式に係わり、特に処理時間の長いタスクが発生した場合にも割り込み毎に行うべきモニタの通常処理を支障なく実行すると共に、処理時間の長いタスクも実行し得るようにした記録装置におけるモニタ制御方式に関するものである。

#### 〔従来の技術〕

近年、複写機やファクシミリ等の記録装置は高

画質、多機能化、高信頼性等進歩がめざましく、各方面に普及されている。しかし、ユーザーからのニーズは多様で、さらに高画質、多機能化、高信頼性であると共に低コスト化、低消費エネルギー化、高速化等の要請に応える必要がある。このような観点から、例えば、複写機を例にとると、システムを複数のサブシステムに分けて各サブシステム毎に必要な処理を行い、全体としてメインシステムで統括管理してシステム全体としての処理の効率化を図っている。

このような各サブシステムあるいは全体を統括するサブシステムにおいては、そのモニタは10 msec単位でタイマ処理、入出力処理、送受信処理、アプリケーションソフト（以下、APPSと言う）のコール処理等を行い、このような10 msec毎のモニタ通常処理を行うことにより全体の制御を円滑に行っている。

#### 〔発明が解決すべき課題〕

ところで、ユーザインタフェースとして、従来のコンソールパネルに代えてCRT画面表示を用

いたものが近年開発され、CRT画面上のタッチ操作で各種モードの選択、設定等を行うようにしており、またジャムが発生した場合のジャム発生位置を順次点灯により知らせるようにしている。そのため、頻繁に画面切換えを行うための処理が発生し、この画面切り換えのために100～200 msec程度の長い処理時間を必要としている。

また、カラー複写機の場合には、濃度設定、コントラスト設定、カラーバランス設定等を行って、各色毎に標準カーブの修正を行うTRC処理のように1000 msec程度の膨大な処理時間を要する計算が必要となっている。

このように処理実行時間が非常にかかってしまうタスクが発生した場合に、その処理のためにCPUが専有されてしまい、割り込み毎に行われるべきモニタの処理ができなくなってしまう。この点について第26図により説明する。

第26図は処理時間の長いタスクが発生した場合のモニタによるタイマ処理における不具合の発生を説明するための図である。

第26図(a)において、モニタは10msec毎の割り込みでタイマ処理、入出力処理、送受信処理、APPSコール処理等の通常の処理をすべて行っている。

今、第26図(b)のタイミング $t_1$ で30msecタイマをセットしたとすると、10msec割り込みによるモニタのタイマ処理で設定値を10msecずつカウントダウンしていく。即ち、タイミング $T_2$ 、 $T_3$ 、 $T_4$ で20msec、10msec、0となってタイムアウトし、これに関連する処理を実行することとなる。このようにモニタの通常処理が10msec以内に終了していれば、特に不具合は生じない。

ところで、第26図(c)に示すように、タイミング $T_1$ において割り込みによりモニタがタイマ処理を行った直後に処理時間の長いAPPSに制御を渡してしまい、APPSがタイミング $T_1$ ～ $T_4$ までCPUを専有し、さらにモニタのタイマ処理直後のタイミング $t_1$ に30msecタイマがセットされたとする。なお、モニタのタイマ処理

はタイマ値をセットする部分、カウントダウンする部分、カウントダウンした結果タイマがタイムアウトした時にその処理を実行する部分とに分かれており、タイマに対して登録する処理を呼べば自由に設定可能である。

こうしてAPPSがCPUをずっと専有してしまうと、タイマはカウントダウン処理を実行することができず、その結果、APPSの処理が終了する $T_4$ のタイミングで初めてカウントダウンが行われて内容は20msecとなり、次にタイミング $T_4$ で10msec、タイミング $T_5$ でタイムアウトとなり、結局タイマセットから60msec後にタイムアウトし、30msecの遅れを生ずることになる。

このようにモニタがAPPSに制御を渡すと、APPSは時間の観念がないので、短い処理でも長い処理でも終了するまで実行してCPUを専有するので、この間モニタは何もできないことになる。

ところで、この10msecという時間は少しでも狂ってはいけないので、最優先レベルの処理とし

て確実に10msecという時間が生成されている。そこで、10msec経過したという要因を残しておく、カウントダウンできるタイミングがきたら、その間の割り込み回数を登録タイマ値と比較して差し引くようにし、結果が0かマイナスの場合にはタイムアウトしたと判断して直ぐにタイムアウト処理を実行するようにすれば、オートクリアタイマ、オートパワーオフタイマ等、非常に長いタイマについては正しいタイマ処理を実行することが可能である。しかし、処理時間の長いAPPSが行われていて、その間モニタのタイマ処理がまわってこない場合にAPPSの終了間際にセットされた設定時間の短いタイマの場合は、長いAPPSの処理時間に加算された10msec経過の要因が一挙に差し引きされるため、直ぐにタイムアウトしてしまい正しいタイマ処理を行うことができない。

本発明は上記課題を解決するためのものである。

本発明の主な目的は、10msecを越える処理時間の長いタスク処理がある場合にも割り込みによ

るモニタの通常処理を支障なく行えるようにすることである。

本発明の他の目的は、非常に長い処理時間を要するタスクをモニタの通常処理を阻害することなく実行できるようにすることである。

本発明の他の目的は、CPUの稼働率を向上させることである。

本発明の他の目的は、分散CPU方式における通信処理を効率的に行うことである。

〔課題を解決するための手段及び作用〕

本発明は、第1図に示すように、所定間隔毎の割り込みにより処理を実行する割り込み処理部1と、バックグラウンド処理部2とを有し、バックグラウンド処理部2は常時バックグラウンド処理を行うと共に、割り込みにより処理を中断し、バックグラウンド処理中断時にモニタの通常処理を行うようにしたこと、またコンテンションバス方式の高速通信と、ポーリング方式の低速通信とを行う通信処理部3を有することを特徴とする。

本発明においては、非常に長い処理時間を要す

るIPSの計算処理、TRC変換テーブルの作成、CRTの画面切換え等を優先レベルの低い処理、すなわちバックグラウンド処理とし、タイマ処理、入出力処理、送受信処理等定期的に必ず行わなければならない処理をレベルの高い処理とし、レベルの低いバックグラウンドの処理は10msec毎の割り込みがある毎に中断してレベルの高い処理を実行し、レベルの高い処理が終了したらバックグラウンド処理を行うようにする。こうして、バックグラウンド処理はモニタの通常処理が行われていないCPUの空き時間を利用して行うようにしたので、どのように長いタスクが発生しても、モニタの通常の処理は支障なく行うことができ、かつ処理の長いタスクはCPUの空き時間を利用して行うようにしているので、その処理を支障なく行うことが可能となる。

また、コンテンションバス方式の高速通信と、ポーリング方式の低速通信とを使用し、システムとIIT、IPSとの時間的余裕のある通信はポーリング方式を使用し、MCB、UIとの短時間

5、ユーザインタフェース(U/I)36から構成され、オプションとして、エディットパッド61、オートドキュメントフィーダ(ADF)62、ソータ63およびフィルムプロジェクタ(F/P)64を備える。

前記IIT、IOT、U/I等の制御を行うためには電気的ハードウェアが必要であるが、これらのハードウェアは、IIT、IITの出力信号をイメージ処理するIPS、U/I、F/P等の各処理の単位毎に複数の基板に分けられており、更にそれらを制御するSYS基板、およびIOT、ADF、ソータ等を制御するためのMCB基板(マシンコントロールボード)等と共に電気制御系収納部33に収納されている。

IIT32は、イメージングユニット37、該ユニットを駆動するためのワイヤ38、駆動ブリー39等からなり、イメージングユニット37内のCCDラインセンサ、カラーフィルタを用いて、カラー原稿を光の原色B(青)、G(緑)、R(赤)毎に読取り、デジタル画像信号に変換して

で処理する必要のある通信はコンテンションバス方式の高速通信で行うことにより、分散CPU方式による特徴を最大限に活かし、かつ効率的に通信処理を行うことができる。

#### 〔実施例〕

本発明が適用される複写機の全体構成の概要を説明する項であって、その構成の中で本発明の実施例を説明する項が(Ⅲ)である。

#### (Ⅰ-1) 装置構成

#### (Ⅰ-2) 電気系制御システムの構成

#### (Ⅱ) システム

#### (Ⅲ) モニタ制御方式

#### (Ⅰ-1) 装置構成

第2図は本発明が適用されるカラー複写機の全体構成の1例を示す図である。

本発明が適用されるカラー複写機は、基本構成となるベースマシン30が、上面に原稿を載置するプラチングガラス31、イメージ入力ターミナル(IIT)32、電気系制御収納部33、イメージ出力ターミナル(IOT)34、用紙トレイ3

IOTへ出力する。

IPSでは、前記IIT32のB、G、R信号をトナーの原色Y(イエロー)、C(シアン)、M(マゼンタ)、K(ブラック)に変換し、さらに、色、階調、精細度等の再現性を高めるために、種々のデータ処理を施してプロセスカラーの階調トナー信号をオン/オフの2値化トナー信号に変換し、IOT34に出力する。

IOT34は、スキャナ40、感材ベルト41を有し、レーザ出力部40aにおいて前記IPSからの画像信号を光信号に変換し、ポリゴンミラー40b、F/θレンズ40cおよび反射ミラー40dを介して感材ベルト41上に原稿画像に対応した潜像を形成させる。感材ベルト41は、駆動ブリー41aによって駆動され、その周囲にクリーナ41b、帯電器41c、Y、M、C、Kの各現像器41dおよび転写器41eが配置されている。そして、この転写器41eに対向して転写装置42が設けられていて、用紙トレイ35から用紙搬送路35aを経て送られる用紙をくわえ込

み、例えば、4色フルカラーコピーの場合には、転写装置42を4回転させ、用紙にY、M、C、Kの順序で転写させる。転写された用紙は、転写装置42から真空搬送装置43を経て定着器45で定着され、排出される。また、用紙搬送路35aには、SSI（シングルシートインサータ）35bからも用紙が選択的に供給されるようになっている。

U/I36は、ユーザが所望の機能を選択してその実行条件を指示するものであり、カラーディスプレイ51と、その横にハードコントロールパネル52を備え、さらに赤外線タッチボード53を組み合わせて画面のソフトボタンで直接指示できるようにしている。

次にベースマシン30へのオプションについて説明する。1つはプラテンガラス31上に、座標入力装置であるエディットパッド61を載置し、入力ペンまたはメモカードにより、各種画像編集を可能にする。また、既存のADF62、ソータ63の取付を可能にしている。

フリーなUIを構築しようとするデータ量が増える。

これに対して、大容量のメモリを搭載したCPUを使用することはできるが、基板が大きくなるので複写機本体に収納するのが困難である、仕様の変更に対して柔軟な対応が困難である、コストが高くなる、等の問題がある。

そこで、本複写機においては、CRTコントローラ等の他の機種あるいは装置との共通化が可能な技術をリモートとしてCPUを分散させることでデータ量の増加に対応するようにしたのである。

電気系のハードウェアは第3図に示されているように、UI系、SYS系およびMCB系の3種の系に大別されている。UI系はUIリモート70を含み、SYS系においては、F/Pの制御を行うF/Pリモート72、原稿読み取りを行うIITリモート73、種々の画像処理を行うIPSリモート74を分散している。IITリモート73はイメージングユニットを制御するためのIITコントローラ73aと、読み取った画像信号を

さらに、本実施例における特徴は、プラテンガラス31上にミラーユニット(M/U)65を載置し、これにF/P64からフィルム画像を投射させ、IIT32のイメージングユニット37で画像信号として読取ることにより、カラーフィルムから直接カラーコピーをとることを可能にしている。対象原稿としては、ネガフィルム、ポジフィルム、スライドが可能であり、オートフォーカス装置、補正フィルタ自動交換装置を備えている。

#### (1-2) 電気系制御システムの構成

この項では、本複写機の電氣的制御システムとして、ハードウェアアーキテクチャー、ソフトウェアアーキテクチャーおよびステート分割について説明する。

##### (A) ハードウェアアーキテクチャーおよびソフトウェアアーキテクチャー

本複写機のようにUIとしてカラーCRTを使用すると、モノクロのCRTを使用する場合に比較してカラー表示のためのデータが増え、また、表示画面の構成、画面遷移を工夫してよりフレン

デジタル化してIPSリモート74に送るVIDEO回路73bを有し、IPSリモート74と共にVCPU74aにより制御される。前記及び後述する各リモートを統括して管理するものとしてSYS(System)リモート71が設けられている。

SYSリモート71はUIの画面遷移をコントロールするためのプログラム等のために膨大なメモリ容量を必要とするので、16ビットマイクロコンピュータを搭載した8086を使用している。なお、8086の他に例えば68000等を使用することもできるものである。

また、MCB系においては、感材ベルトにレーザで潜像を形成するために使用するビデオ信号をIPSリモート74から受け取り、IOTに送出するためのラスタ出力スキャン(Raster Output Scan: ROS)インターフェースであるVCB(Video Control Board)リモート76、転写装置(タートル)のサーボのためのRCBリモート77、更にはIOT、ADF、ソータ、アクセサリのためのI/OポートとしてのIOBRモ

ト78、およびアクセサリリモート79を分散させ、それらを統括して管理するためにMCB (Master Control Board) リモート75が設けられている。

なお、図中の各リモートはそれぞれ1枚の基板で構成されている。また、図中の太い実線は187.5 kbpsのLNET高速通信網、太い破線は9600bpsのマスター/スレーブ方式シリアル通信網をそれぞれ示し、細い実線はコントロール信号の伝送路であるホットラインを示す。また、図中76.8kbpsとあるのは、エディットパッドに描かれた図形情報、メモリカードから入力されたコピーモード情報、編集領域の図形情報をUIリモート70からIPSリモート74に通知するための専用回線である。更に、図中CCC (Communication Control Chip) とあるのは、高速通信回線LNETのプロトコルをサポートするICである。

以上のようにハードウェアアーキテクチャーは、UI系、SYS系、MCB系の3つに大別されるが、これらの処理の分担を第4図のソフトウェア

アーキテクチャーを参照して説明すると次のようである。なお、図中の矢印は第3図に示す187.5 kbpsのLNET高速通信網、9600bpsのマスター/スレーブ方式シリアル通信網を介して行われるデータの授受またはホットラインを介して行われる制御信号の伝送関係を示している。

UIリモート70は、LLUI (Low Level UI) モジュール80と、エディットパッドおよびメモリカードについての処理を行うモジュール (図示せず) から構成されている。LLUIモジュール80は通常CRTコントローラとして知られているものと同様であって、カラーCRTに画面を表示するためのソフトウェアモジュールであり、その時々でどのような絵の画面を表示するかは、SYSUIモジュール81またはMCBUIMジュール86により制御される。これによりUIリモートを他の機種または装置と共通化することができるとは明かである。なぜなら、どのような画面構成とするか、画面遷移をどうするかは機種によって異なるが、CRTコントローラはC

RTと一体で使用されるものであるからである。

SYSリモート71は、SYSUIモジュール81と、SYSTEMモジュール82、およびSYS.DIAGモジュール83の3つのモジュールで構成されている。

SYSUIモジュール81は画面遷移をコントロールするソフトウェアモジュールであり、SYSTEMモジュール82は、どの画面でソフトパネルのどの座標が選択されたか、つまりどのようなジョブが選択されたかを認識するF/F (Feature Function) 選択のソフトウェア、コピー実行条件に矛盾が無いかどうか等最終的にジョブをチェックするジョブ確認のソフトウェア、および、他のモジュールとの間でF/F選択、ジョブリカバリ、マシンステート等の種々の情報の授受を行うための通信を制御するソフトウェアを含むモジュールである。

SYS.DIAGモジュール83は、自己診断を行うダイアグノスティックステートでコピー動作を行うカスタマーシミュレーションモードの場

合に動作するモジュールである。カスタマーシミュレーションモードは通常のコピーと同じ動作をするので、SYS.DIAGモジュール83は実質的にはSYSTEMモジュール82と同じなのであるが、ダイアグノスティックという特別なステートで使用されるので、SYSTEMモジュール82とは別に、しかし一部が重畳されて記載されているものである。

また、IITリモート73にはイメージングユニットに使用されているステッピングモータの制御を行うIITモジュール84が、IPSリモート74にはIPSに関する種々の処理を行うIPSモジュール85がそれぞれ格納されており、これらのモジュールはSYSTEMモジュール82によって制御される。

一方、MCBリモート75には、ダイアグノスティック、オーディترون (Auditron) およびジャム等のフォールトの場合に画面遷移をコントロールするソフトウェアであるMCBUIMジュール86、感材ベルトの制御、現像機の制御、フュ

ーザの制御等コピーを行う際に必要な処理を行う IOTモジュール90、ADFを制御するための ADFモジュール91、ソータを制御するための SORTERモジュール92の各ソフトウェアモジュールとそれらを管理するコピアエグゼクティブモジュール87、および各種診断を行うダイアグエグゼクティブモジュール88、暗唱番号で電子カウンターにアクセスして料金処理を行うオーディトロンモジュール89を格納している。

また、RCBリモート77には転写装置の動作を制御するタートルサーボモジュール93が格納されており、当該タートルサーボモジュール93はゼログラフィーサイクルの転写工程を司るために、IOTモジュール90の管理の下に置かれている。なお、図中、コピアエグゼクティブモジュール87とダイアグエグゼクティブモジュール88が重複しているのは、SYSTEMモジュール82とSYS. DIAGモジュール83が重複している理由と同様である。

以上の処理の分担をコピー動作に従って説明す

第4図に示されているように、IOTモジュール90とIITモジュール84の間ではPR-TRUEという信号と、LE@REGという2つの信号のやり取りが行われる。具体的にいえば、IOTの制御の基準タイミングであるPR (PITCH RESET) 信号はMCBより感材ベルトの回転を2または3分割して連続的に発生される。つまり、感材ベルトは、その有効利用とコピースピード向上のために、例えばコピー用紙がA3サイズの場合には2ピッチ、A4サイズの場合には3ピッチというように、使用されるコピー用紙のサイズに応じてピッチ分割されるようになされているので、各ピッチ毎に発生されるPR信号の周期は、例えば2ピッチの場合には3secと長くなり、3ピッチの場合には2secと短くなる。

さて、MCBで発生されたPR信号は、VIDEO信号関係を取り扱うVCBリモート等のIOT内の必要な箇所にホットラインを介して分配される。

VCBはその内部にゲート回路を有し、IOT

と次のようである。コピー動作は現像される色の違いを別にすればよく似た動作の繰り返しであり、第5図(a)に示すようにいくつかのレイヤに分けて考えることができる。

1枚のカラーコピーはピッチと呼ばれる最小の単位を何回か繰り返すことで行われる。具体的には、1色のコピーを行うについて、現像機、転写装置等をどのように動作させるか、ジャムの検知はどのように行うか、という動作であって、ピッチ処理をY、M、Cの3色について行えば3色カラーのコピーが、Y、M、C、Kの4色について行えば4色フルカラーのコピーが1枚出来上がることになる。これがコピーレイヤであり、具体的には、用紙に各色のトナーを転写した後、フューザで定着させて複写機本体から排紙する処理を行うレイヤである。ここまでの処理の管理はMCB系のコピアエグゼクティブモジュール87が行う。

勿論、ピッチ処理の過程では、SYS系に含まれているIITモジュール84およびIPSモジュール85も使用されるが、そのために第3図、

内でイメージングが可能、即ち、実際に感材ベルトにイメージを露光することが可能なピッチのみ選択的にIPSリモートに対して出力する。この信号がPR-TRUE信号である。なお、ホットラインを介してMCBから受信したPR信号に基づいてPR-TRUE信号を生成するための情報は、LNETによりMCBから通知される。

これに対して、実際に感材ベルトにイメージを露光することができない期間には、感材ベルトには1ピッチ分の空ピッチを作ることになり、このような空ピッチに対してはPR-TRUE信号は出力されない。このようなPR-TRUEが発生されないピッチとしては、例えば、転写装置での転写が終了した用紙を排出してから次の用紙を転写装置に供給するまでの間の期間を挙げることができる。つまり、例えば、A3サイズのように長い用紙を最後の転写と共に排出するとすると、用紙の先端がフューザの入口に入ったときのショックで画質が劣化するために一定長以上の用紙の場合には最後の転写が終了してもそのまま排出せず、



後述するグリッパバーで保持したまま一定速度でもう一周回転させた後排出するようになされているため、感材ベルトには1ピッチ分のスキップが必要となるのである。

また、スタートキーによるコピー開始からサイクルアップシーケンスが終了するまでの間もPR-TRUE信号は出力されない。この期間にはまだ原稿の読み取りが行われておらず、従って、感材ベルトにはイメージを露光することができないからである。

VCBリモートから出力されたPR-TRUE信号は、IPSリモートで受信されると共に、そのままIITリモートにも伝送されて、IITのスキャンスタートのためのトリガー信号として使用される。

これによりIITリモート73およびIPSリモート74をIOTに同期させてピッチ処理を行わせることができる。また、このときIPSリモート74とVCBリモート76の間では、感材ベルトに潜像を形成するために使用されるレーザ光

2ピッチのPR信号を出力する。IOT78bのサイクルアップシーケンスが終了すると、その時点からPR信号に同期してPR-TRUE信号が、イメージングが必要なピッチのみに対応してIITコントローラ73aに出力される。

また、IOT78bは、ROS（ラスタアウットブツスキャン）の1ライン分の回転毎に出力されるIOT-LS（ラインシンク）信号を、VCPU74a内のTG（タイミングジェネレータ）に送り、ここでIOT-LSに対してIPSの総パイプライン遅延分だけ見掛け上の位相を進めたIPS-LSをIITコントローラ73aに送る。

IITコントローラ73aは、PR-TRUE信号が入ると、カウンタをイネーブルしてIOT-LS信号をカウントし、所定のカウンタ数に達すると、イメージングユニット37を駆動させるステッピングモータ213の回転をスタートさせてイメージングユニットが原稿のスキャンを開始する。さらにカウントしてT2秒後原稿読取開始

を変調するためのビデオ信号の授受が行われ、VCBリモート76で受信されたビデオ信号は並列信号から直列信号に変換された後、直接ROSへVIDEO変調信号としてレーザ出力部40aに与えられる。

以上の動作が4回繰り返されると1枚の4色フルカラーコピーが出来上がり、1コピー動作は終了となる。

次に、第5図(b)～(e)により、IITで読取られた画像信号をIOTに出力し最終的に転写ポイントで用紙に転写させるまでの信号のやりとりとそのタイミングについて説明する。

第5図(b)、(c)に示すように、SYSリモート71からスタートジョブのコマンドが入ると、IOT78bではメインモータの駆動、高圧電源の立ち上げ等サイクルアップシーケンスに入る。IOT78bは、感材ベルト上に用紙長に対応した潜像を形成させるために、PR（ピッチリセット）信号を出力する。例えば、感材ベルトが1回転する毎に、A4では3ピッチ、A3では

位置でLE@REGを出力しこれをIOT78bに送る。

この原稿読取開始位置は、予め例えば電源オン後1回だけ、イメージングユニットを駆動させてレジサ217の位置（レジ位置の近く、具体的にはレジ位置よりスキャン側に約10mm）を一度検出して、その検出位置を元に真のレジ位置を計算で求め、また同時に通常停止位置（ホームポジション）も計算で求めることができる。また、レジ位置は機械のばらつき等でマシン毎に異なるため、補正値をNVMに保持しておき、真のレジ位置とホームポジションの計算時に補正を行うことにより、正確な原稿読取開始位置を設定することができる。この補正値は工場またはサービスマン等により変更することができ、この補正値を電氣的に書き換えるだけで実施でき、機械的調整は不要である。なお、レジサ217の位置を真のレジ位置よりスキャン側に約10mmずらしているのは、補正を常にマイナス値とし、調整およびソフトを簡単にするためである。

また、IITコントローラ73aは、LE@REGと同期してIMAGE-AREA信号を出力する。このIMAGE-AREA信号の長さは、スキャン長に等しいものであり、スキャン長はSYSTEMモジュール82よりIITモジュール84へ伝達されるスタートコマンドによって定義される。具体的には、原稿サイズを検知してコピーを行う場合には、スキャン長は原稿長さであり、倍率を指定してコピーを行う場合には、スキャン長はコピー用紙長と倍率（100%を1とする）との除数で設定される。IMAGE-AREA信号は、VCPU74aを経由しそこでIIT-PPS（ページシンク）と名前を変えてIPS74に送られる。IIT-PPSはイメージ処理を行う時間を示す信号である。

LE@REGが出力されると、IOT-LS信号に同期してラインセンサの1ライン分のデータが読み取られ、VIDEO回路（第3図）で各種補正処理、A/D変換が行われIPS74に送られる。IPS74においては、IOT-LSと同

nでLE@REGを出力するとき、LE@REGはPR-TRUEに対してT1時間だけ遅れることになる。この遅れは最大1ラインシンク分で、4色フルカラーコピーの場合にはこの遅れが累積してしまい出力画像に色ズレとなって現れてしまう。

そのために、先ず、第5図（c）に示すように、1回目のLE@REGが入ると、カウンタ1がカウントを開始し、2、3回目のLE@REGが入ると、カウンタ2、3がカウントを開始し、それぞれのカウンタが転写位置までのカウント数pに達するとこれをクリアして、以下4回目以降のLE@REGの入力に対して順番にカウンタを使用して行く。そして、第5図（e）に示すように、LE@REGが入ると、IOT-CLKの直前のパルスからの時間T3を補正用クロックでカウントする。感材ベルトに形成された潜像が転写位置に近ずき、IOT-CLKが転写位置までのカウント数pをカウントすると、同時に補正用クロックがカウントを開始し、上記時間T3に相当する

期して1ライン分のビデオデータをIOT78bに送る。このときIOT-BYTE-CLKの反転信号であるRTN-BYTE-CLKをビデオデータと並列してIOTへ送り返しデータとクロックを同様に遅らせることにより、同期を確実にとるようにしている。

IOT78bにLE@REGが入力されると、同様にIOT-LS信号に同期してビデオデータがROSに送られ、感材ベルト上に潜像が形成される。IOT78bは、LE@REGが入るとそのタイミングを基準にしてIOT-CLKによりカウントを開始し、一方、転写装置のサーボモータは、所定カウント数の転写位置で用紙の先端がくるように制御される。ところで、第5図（d）に示すように、感材ベルトの回転により出力されるPR-TRUE信号とROSの回転により出力されるIOT-LS信号とはもともと同期していない。このため、PR-TRUE信号が入り次のIOT-LSからカウントを開始し、カウントmでイメージングユニット37を動かし、カウント

カウント数rを加えた点が、正確な転写位置となり、これを転写装置の転写位置（タイミング）コントロール用カウンタの制御に上乗せし、LE@REGの入力に対して用紙の先端が正確に同期するように転写装置のサーボモータを制御している。

以上がコピーレイヤまでの処理であるが、その上に、1枚の原稿に対してコピー単位のジョブを何回行うかというコピー枚数を設定する処理があり、これがパーオリジナル（PER ORIGINAL）レイヤで行われる処理である。更にその上には、ジョブのパラメータを変える処理を行うジョブプログラミングレイヤがある。具体的には、ADFを使用するか否か、原稿の一部の色を変える、偏倍機能を使用するか否か、ということである。これらパーオリジナル処理とジョブプログラミング処理はSYS系のSYSモジュール82が管理する。そのためにSYSTEMモジュール82は、LLUモジュール80から送られてきたジョブ内容をチェック、確定し、必要なデータを作成して、9600bpsシリアル通信網によりIITモジュール

ル 84、IPS モジュール 85 に通知し、また LNET により MCB 系にジョブ内容を通知する。

以上述べたように、独立な処理を行うもの、他の機種、あるいは装置と共通化が可能な処理を行うものをリモートとして分散させ、それらを UI 系、SYS 系、および MCB 系に大別し、コピー処理のレイヤに従ってマシンを管理するモジュールを定めたので、設計者の業務を明確にできる、ソフトウェア等の開発技術を均一化できる、納期およびコストの設定を明確化できる、仕様の変更等があった場合にも関係するモジュールだけを変更することで容易に対応することができる、等の効果が得られ、以て開発効率を向上させることができるものである。

#### (B) ステート分割

以上、UI 系、SYS 系および MCB 系の処理の分担について述べたが、この項では UI 系、SYS 系、MCB 系がコピー動作のその時々でどのような処理を行っているかをコピー動作の順を追って説明する。

パーソリジナル処理およびジョブプログラミング処理は SYS モジュール 82 で管理されるというように処理が分担されているから、これに対応して各ステートにおいて SYS モジュール 82、コピアエグゼクティブモジュール 87 のどちらが全体のコントロール権を有するか、また、UI マスター権を有するかが異なるのである。第 6 図においては縦線で示されるステートは UI マスター権を MCB 系のコピアエグゼクティブモジュール 87 が有することを示し、黒く塗りつぶされたステートは UI マスター権を SYS モジュール 82 が有することを示している。

第 6 図に示すステート分割の内パワー ON からスタンバイまでを第 7 図を参照して説明する。

電源が投入されてパワー ON になされると、第 3 図で SYS リモート 71 から IIT リモート 73 および IPS リモート 74 に供給される IPS リセット信号および IIT リセット信号が H (HIGH) となり、IPS リモート 74、IIT リモート 73 はリセットが解除されて動作を開始する。

複写機では、パワー ON からコピー動作、およびコピー動作終了後の状態をいくつかのステートに分割してそれぞれのステートで行うジョブを決めておき、各ステートでのジョブを全て終了しなければ次のステートに移行しないようにしてコントロールの能率と正確さを期するようにしている。これをステート分割といい、本複写機においては第 6 図に示すようなステート分割がなされている。

本複写機におけるステート分割で特徴的なことは、各ステートにおいて、当該ステート全体を管理するコントロール権および当該ステートで UI を使用する UI マスター権が、あるときは SYS リモート 71 にあり、またあるときは MCB リモート 75 にあることである。つまり、上述したように CPU を分散させたことによって、UI リモート 70 の LLUI モジュール 80 は SYS UI モジュール 81 ばかりでなく MCB UI モジュール 86 によっても制御されるのであり、また、ピッチおよびコピー処理は MCB 系のコピアエグゼクティブモジュール 87 で管理されるのに対して、

また、電源電圧が正常になったことを検知するとパワーノーマル信号が立ち上がり、MCB リモート 75 が動作を開始し、コントロール権および UI マスター権を確立すると共に、高速通信網 LNET のテストを行う。また、パワーノーマル信号はホットラインを通じて MCB リモート 75 から SYS リモート 71 に送られる。

MCB リモート 75 の動作開始後所定の時間 T0 が経過すると、MCB リモート 75 からホットラインを通じて SYS リモート 71 に供給されるシステムリセット信号が H となり、SYS リモート 71 のリセットが解除されて動作が開始されるが、この際、SYS リモート 71 の動作開始は、SYS リモート 71 の内部の信号である 86 NM I、86 リセットという二つの信号により上記 T0 時間の経過後更に 200  $\mu$ sec 遅延される。この 200  $\mu$ sec という時間は、クラッシュ、即ち電源の瞬断、ソフトウェアの暴走、ソフトウェアのバグ等による一過性のトラブルが生じてマシンが停止、あるいは暴走したときに、マシンがどのステ

ートにあるかを不揮発性メモリに格納するために設けられているものである。

SYSリモート71が動作を開始すると、約3.8secの間コアテスト、即ちROM、RAMのチェック、ハードウェアのチェック等を行う。このとき不所望のデータ等が入力されると暴走する可能性があるので、SYSリモート71は自らの監督下で、コアテストの開始と共にIPSリセット信号およびIITリセット信号をL(Low)とし、IPSリモート74およびIITリモート73をリセットして動作を停止させる。

SYSリモート71は、コアテストが終了すると、10〜3100msecの間CCCセルフテストを行うと共に、IPSリセット信号およびIITリセット信号をHとし、IPSリモート74およびIITリモート73の動作を再開させ、それぞれコアテストを行わせる。CCCセルフテストは、LNETに所定のデータを送出して自ら受信し、受信したデータが送信されたデータと同じであることを確認することで行う。なお、CCCセルフテ

ストを行うについては、セルフテストの時間が重ならないように各CCCに対して時間が割り当てられている。

つまり、LNETにおいては、SYSリモート71、MCBリモート75等の各ノードはデータを送信したいときに送信し、もしデータの衝突が生じていれば所定時間経過後再送信を行うというコンテンション方式を採用しているので、SYSリモート71がCCCセルフテストを行っているとき、他のノードがLNETを使用しているとデータの衝突が生じてしまい、セルフテストが行えないからである。従って、SYSリモート71がCCCセルフテストを開始するときには、MCBリモート75のLNETテストは終了している。

CCCセルフテストが終了すると、SYSリモート71は、IPSリモート74およびIITリモート73のコアテストが終了するまで待機し、T1の期間にSYSTEMノードの通信テストを行う。この通信テストは、9600bpsのシリアル通信網のテストであり、所定のシーケンスで所定

のデータの送受信が行われる。当該通信テストが終了すると、T2の期間にSYSリモート71とMCBリモート75の間でLNETの通信テストを行う。即ち、MCBリモート75はSYSリモート71に対してセルフテストの結果を要求し、SYSリモート71は当該要求に応じてこれまで行ってきたテストの結果をセルフテストリザルトとしてMCBリモート75に発行する。

MCBリモート75は、セルフテストリザルトを受け取るとトークンバスをSYSリモート71に発行する。トークンバスはUIマスター権をやり取りする札であり、トークンバスがSYSリモート71に渡されることで、UIマスター権はMCBリモート75からSYSリモート71に移ることになる。ここまではパワーオンシーケンスである。当該パワーオンシーケンスの期間中、UIリモート70は「しばらくお待ち下さい」等の表示を行うと共に、自らのコアテスト、通信テスト等、各種のテストを行う。

上記のパワーオンシーケンスの内、セルフテ

ストリザルトの要求に対して返答されない、またはセルフテストリザルトに異常がある場合には、MCBリモート75はマシンをデッドとし、UIコントロール権を発動してUIリモート70を制御し、異常が生じている旨の表示を行う。これがマシンデッドのステートである。

パワーオンステートが終了すると、次に各リモートをセットアップするためにイニシャライズステートに入る。イニシャライズステートではSYSリモート71が全体のコントロール権とUIマスター権を有している。従って、SYSリモート71は、SYS系をイニシャライズすると共に、「INITIALIZE SUBSYSTEM」コマンドをMCBリモート75に発行してMCB系をもイニシャライズする。その結果はサブシステムステータス情報としてMCBリモート75から送られてくる。これにより例えばIOTではフューズを加熱したり、トレイのエレベータが所定の位置に配置されたりしてコピーを行う準備が整えられる。ここまではイニシャライズステートである。

イニシャライズが終了すると各リモートは待機状態であるスタンバイに入る。この状態においてもUIマスター権はSYSリモート71が有しているので、SYSリモート71はUIマスター権に基づいてUI画面上にF/Fを表示し、コピー実行条件を受け付ける状態に入る。このときMCBリモート75はIOTをモニターしている。また、スタンバイ状態では、異常がないかどうかをチェックするためにMCBリモート75は、500msec 毎にバックグラウンドボールをSYSリモート71に発行し、SYSリモート71はこれに対してセルフテストリザルトを200msec 以内にMCBリモート75に返すという処理を行う。このときセルフテストリザルトが返ってこない、あるいはセルフテストリザルトの内容に異常があるときには、MCBリモート75はUIリモート70に対して異常が発生した旨を知らせ、その旨の表示を行わせる。

スタンバイ状態においてオーディترونが使用されると、オーディترونステートに入り、

トアップに入り、各リモートでは指定されたジョブを行うための前準備を行う。例えば、IOTモジュール90ではメインモータの駆動、感材ベルトのパラメータの合わせ込み等が行われる。スタートジョブに対する応答であるACK (Acknowledge) がMCBリモート75から送り返されたことを確認すると、SYSリモート71は、IITリモート73にブリスキャンを行わせる。ブリスキャンには、原稿サイズを検出するためのブリスキャン、原稿の指定された位置の色を検出するためのブリスキャン、塗り絵を行う場合の閉ループ検出のためのブリスキャン、マーカ編集の場合のマーカ読み取りのためのブリスキャンの4種類があり、選択されたF/Fに応じて最高3回までブリスキャンを行う。このときUIには例えば「しばらくお待ち下さい」等の表示が行われる。

ブリスキャンが終了すると、IITレディというコマンドが、コピアエグゼクティブモジュール87に発行され、ここからサイクルアップに入る。サイクルアップは各リモートの立ち上がり時間を

MCBリモート75はオーディترونコントロールを行うと共に、UIリモート70を制御してオーディترونのための表示を行わせる。スタンバイ状態においてF/Fが設定され、スタートキーが押されるとプログレスステートに入る。プログレスステートは、セットアップ、サイクルアップ、ラン、スキップピッチ、ノーマルサイクルダウン、サイクルダウンシャットダウンという6ステートに細分化されるが、これらのステートを、第8図を参照して説明する。

第8図は、プラテンモード、4色フルカラー、コピー設定枚数3の場合のタイミングチャートを示す図である。

SYSリモート71は、スタートキーが押されたことを検知すると、ジョブの内容をシリアル通信網を介してIITリモート73およびIPSリモート74に送り、またLNETを介してジョブの内容をスタートジョブというコマンドと共にMCBリモート75内のコピアエグゼクティブモジュール87に発行する。このことでマシンはセッ

待ち合わせる状態であり、MCBリモート75はIOT、転写装置の動作を開始し、SYSリモート71はIPSリモート74を初期化する。このときUIは、現在プログレスステートにあること、および選択されたジョブの内容の表示を行う。

サイクルアップが終了するとランに入り、コピー動作が開始されるが、先ずMCBリモート75のIOTモジュール90から1個目のPR0が出されるとIITは1回目のスキャンを行い、IOTは1色目の現像を行い、これで1ピッチの処理が終了する。次に再びPR0が出されると2色目の現像が行われ、2ピッチ目の処理が終了する。この処理を4回繰り返して、4ピッチの処理が終了するとIOTはフューザでトナーを定着し、排紙する。これで1枚目のコピー処理が完了する。以上の処理を3回繰り返すと3枚のコピーができる。

ピッチレイヤの処理およびコピーレイヤの処理はMCBリモート75が管理するが、その上のレイヤであるバーオリジナルレイヤで行うコピー設定枚数の処理はSYSリモート71が行う。従っ

て、現在何枚目のコピーを行っているかをSYSリモート71が認識できるように、各コピーの1個目のPROが出されるとき、MCBリモート75はSYSリモート71に対してメイドカウント信号を発行するようになされている。また、最後のPROが出されるときには、MCBリモート75はSYSリモート71に対して「RDY FOR NEXT JOB」というコマンドを発行して次のジョブを要求する。このときスタートジョブを発行するとジョブを続行できるが、ユーザが次のジョブを設定しなければジョブは終了であるから、SYSリモート71は「END JOB」というコマンドをMCBリモート75に発行する。MCBリモート75は「END JOB」コマンドを受信してジョブが終了したことを確認すると、マシンはノーマルサイクルダウンに入る。ノーマルサイクルダウンでは、MCBリモート75はIOTの動作を停止させる。

サイクルダウンの途中、MCBリモート75は、コピーされた用紙が全て排紙されたことが確認さ

れるとその旨を「DELIVERED JOB」コマンドでSYSリモート71に知らせ、また、ノーマルサイクルダウンが完了してマシンが停止すると、その旨を「IOT STAND BY」コマンドでSYSリモート71に知らせる。これによりプログレスステートは終了し、スタンバイステートに戻る。

なお、以上の例ではスキップピッチ、サイクルダウンシャットダウンについては述べられていないが、スキップピッチにおいては、SYSリモート71はSYS系を次のジョブのためにイニシャライズし、また、MCBリモート75では次のコピーのために待機している。また、サイクルダウンシャットダウンはフォールトの際のステートであるので、当該ステートにおいては、SYSリモート71およびMCBリモート75は共にフォールト処理を行う。

以上のようにプログレスステートにおいては、MCBリモート75はピッチ処理およびコピー処理を管理し、SYSリモート71はバーオリジナル

ル処理およびジョブプログラミング処理を管理しているので、処理のコントロール権は双方が処理の分担に応じてそれぞれ有している。これに対してUIマスター権はSYSリモート71が有している。なぜなら、UIにはコピーの設定枚数、選択された編集処理などを表示する必要があり、これらはバーオリジナル処理もしくはジョブプログラミング処理に属し、SYSリモート71の管理下に置かれるからである。

プログレスステートにおいてフォールトが生じるとフォールトリカバリーステートに移る。フォールトというのは、ノーペーパー、ジャム、部品の故障または破損等マシンの異常状態の総称であり、F/Fの再設定等を行うことでユーザがリカバリーできるものと、部品の交換などサービスマンがリカバリーしなければならないものの2種類がある。上述したように基本的にはフォールトの表示はMCBUIモジュール86が行うが、F/FはSYSモジュール82が管理するので、F/Fの再設定でリカバリーできるフォールトに関し

てはSYSモジュール82がリカバリーを担当し、それ以外のリカバリーに関してはコピアエグゼクティブモジュール87が担当する。

また、フォールトの検出はSYS系、MCB系それぞれに行われる。つまり、IIT、IPS、F/PはSYSリモート71が管理しているのでSYSリモート71が検出し、IOT、ADP、ソータはMCBリモート75が管理しているのでMCBリモート75が検出する。従って、本複写機においては次の4種類のフォールトがあることが分かる。

①SYSノードで検出され、SYSノードがリカバリーする場合

例えば、F/Pが準備されないままスタートキーが押されたときにはフォールトとなるが、ユーザは再度F/Fを設定することでリカバリーできる。

②SYSノードで検出され、MCBノードがリカバリーする場合

この種のフォールトには、例えば、レジセンサ



合に用いるモードであり、カスタマーシミュレーションモードは、通常ユーザがコピーする場合に使用するカスタマーモードをダイアグで使用するモードである。

いま、カスタマーモードのスタンバイステートから所定の操作により図のAのルートによりTECH REPモードに入ったとする。TECH REPモードで各種のチェック、パラメータの設定、モードの設定を行っただけで終了し、再びカスタマーモードに戻る場合（図のBのルート）には所定のキー操作を行えば、第6図に示すようにパワーオンのステートに移り、第7図のシーケンスによりスタンバイステートに戻ることができるが、本複写機はカラーコピーを行い、しかも種々の編集機能を備えているので、TECH REPモードで種々のパラメータの設定を行った後に、実際にコピーを行ってユーザが要求する色が出るかどうか、編集機能は所定の通りに機能するかどうか等を確認する必要がある。これを行うのがカスタマーシミュレーションモードであり、ピリン

グを行わない点、UIにはダイアグである旨の表示がなされる点でカスタマーモードと異なっている。これがカスタマーモードをダイアグで使用するカスタマーシミュレーションモードの意味である。なお、TECH REPモードからカスタマーシミュレーションモードへの移行（図のCのルート）、その逆のカスタマーシミュレーションモードからTECH REPモードへの移行（図のDのルート）はそれぞれ所定の操作により行うことができる。また、TECH REPモードはダイアグエグゼクティブモジュール88（第4図）が行うのでコントロール権、UIマスター権は共にMCBノードが有しているが、カスタマーシミュレーションモードはSYS、DIAGモジュール83（第4図）の制御の基で通常のコピー動作を行うので、コントロール権、UIマスター権は共にSYSノードが有する。

## （Ⅱ）システム

### （Ⅱ-1）システムの位置付け

第10図はシステムと他のリモートとの関係を

示す図である。

前述したように、リモート71にはSYSUIモジュール81とSYSTEMモジュール82が搭載され、SYSUI81とSYSTEMモジュール82間はモジュール間インタフェースによりデータの授受が行われ、またSYSTEMモジュール82とIIT73、IPS74との間はリアル通信インターフェースで接続され、MCB75、ROS76、RAIB79との間はLNET高速通信網で接続されている。

次にシステムのモジュール構成について説明する。

### （Ⅱ-2）システムのモジュール構成

第11図はシステムのモジュール構成を示す図である。

本複写機においては、IIT、IPS、IOT等の各モジュールは部品のように考え、これらをコントロールするシステムの各モジュールは頭脳を持つように考えている。そして、分散CPU方式を採用し、システム側ではバーオリジナル処理

およびジョブプログラミング処理を担当し、これに対応してイニシャライズステート、スタンバイステート、セットアップステート、サイクルステートを管理するコントロール権、およびこれらのステートでUIを使用するUIマスター権を有しているので、それに対応するモジュールでシステムを構成している。

システムメイン100は、SYSUIやMCB等からの受信データを内部バッファに取り込み、また内部バッファに格納したデータをクリアし、システムメイン100の下位の各モジュールをコールして処理を渡し、システムステートの更新処理を行っている。

M/Cイニシャライズコントロールモジュール101は、パワーオンしてからシステムがスタンバイ状態になるまでのイニシャライズシーケンスをコントロールしており、MCBによるパワーオン後の各種テストを行うパワーオン処理が終了すると起動される。

M/Cセットアップコントロールモジュール1



03はスタートキーが押されてから、コピーレイアウトの処理を行うMCBを起動するまでのセットアップシーケンスをコントロールし、具体的にはSYSUIから指示されたFEATURE（使用者の要求を達成するためのM/Cに対する指示項目）に基づいてジョブモードを作成し、作成したジョブモードに従ってセットアップシーケンスを決定する。

第12図(a)に示すように、ジョブモードの作成は、F/Fで指示されたモードを解析し、ジョブを切り分けている。この場合ジョブとは、使用者の要求によりM/Cがスタートしてから要求通りのコピーが全て排出され、停止されるまでのM/C動作を言い、使用者の要求に対して作業分割できる最小単位、ジョブモードの集合体である。例えば、嵌め込み合成の場合で説明すると、第12図(b)示すように、ジョブモードは削除と移動、抽出とからなり、ジョブはこれらのモードの集合体となる。また、第12図(c)に示すようにADF原稿3枚の場合においては、ジョブモードはそれぞれ

また、M/C停止中、あるいは動作中に発生するスルーコマンドを相手先リモートに通知する機能を果たしている。

フォールトコントロールモジュール106はIIT、IPSからの立ち下げ要因を監視し、要因発生時にMCBに対して立ち下げ要求し、具体的にはIIT、IPSからのフェイルコマンドによる立ち下げを行い、またMCBからの立ち下げ要求が発生後、M/C停止時のリカバリーを判断して決定し、例えばMCBからのジャムコマンドによりリカバリーを行っている。

コミュニケーションコントロールモジュール107はIITからのIITレディ信号の設定、イメージエリアにおける通信のイネーブル/ディスエイブルを設定している。

DIAGコントロールモジュール108は、DIAGモードにおいて、入力チェックモード、出力チェックモード中のコントロールを行っている。

#### (II-3) システムのデータ授受

次に、これらシステムの各モジュール同士、あ

れ原稿1、原稿2、原稿3に対するフィード処理であり、ジョブはそれらの集合となる。

そして、自動モードの場合はドキュメントスキャン、ぬり検モードの時はブレスキャン、マーカー編集モードの時はブレスキャン、色検知モードの時はサンプルスキャンを行い（ブレスキャンは最高3回）、またコピーサイクルに必要なコピーモードをIIT、IPS、MCBに対して配付し、セットアップシーケンス終了時MCBを起動する。

M/Cスタンバイコントロールモジュール102はM/Cスタンバイ中のシーケンスをコントロールし、具体的にはスタートキーの受付、色登録のコントロール、ダイアグモードのエントリー等を行っている。

M/Cコピーサイクルコントロールモジュール104はMCBが起動されてから停止するまでのコピーシーケンスをコントロールし、具体的には用紙フィードカウントの通知、JOBの終了を判断してIITの立ち上げ要求、MCBの停止を判断してIPSの立ち下げ要求を行う。

あるいは他のサブシステムとのデータの授受について説明する。

第13図はシステムと各リモートとのデータフロー、およびシステム内モジュール間データフローを示す図である。図のA~Nはシリアル通信を、Zはホットラインを、①~⑫はモジュール間データを示している。

SYSUIリモートとイニシャライズコントロール部101との間では、SYSUIからはCRTの制御権をSYSTEM NODEに渡すTOKENコマンドが送られ、一方イニシャライズコントロール部101からはコンフィグコマンドが送られる。

SYSUIリモートとスタンバイコントロール部102との間では、SYSUIからはモードチェンジコマンド、スタートコピーコマンド、ジョブキャンセルコマンド、色登録リクエストコマンド、トレイコマンドが送られ、一方スタンバイコントロール部102からはM/Cステータスコマンド、トレイステータスコマンド、トナーステ

タスコマンド、回収ボトルステータスコマンド、色登録ANSコマンド、TOKENコマンドが送られる。

SY SUIリモートとセットアップコントロール部103との間では、セットアップコントロール部103からはM/Cステータスコマンド（プログレス）、APMSステータスコマンドが送られ、一方SY SUIリモートからはストップリクエストコマンド、インターラプトコマンドが送られる。

IPSリモートとイニシャライズコントロール部101との間では、IPSリモートからはイニシャライズエンドコマンドが送られ、イニシャライズコントロール部101からはNVMパラメータコマンドが送られる。

II Tリモートとイニシャライズコントロール部101との間では、II TリモートからはII Tレディコマンド、イニシャライズコントロール部101からはNVMパラメータコマンド、INITIALIZEコマンドが送られる。

ライズコマンドが送られる。

II Tリモートとセットアップコントロール部103との間では、II TリモートからはII Tレディコマンド、イニシャライズエンドコマンドが送られ、セットアップコントロール部103からはドキュメントスキャンスタートコマンド、サンプルスキャンスタートコマンド、コピースキャンスタートコマンドが送られる。

MCBリモートとスタンバイコントロール部102との間では、スタンバイコントロール部102からイニシャライズサブシステムコマンド、スタンバイセレクションコマンドが送られ、MCBリモートからはサブシステムステータスコマンドが送られる。

MCBリモートとセットアップコントロール部103との間では、セットアップコントロール部103からスタートジョブコマンド、II Tレディコマンド、ストップジョブコマンド、デクレアシステムフォールトコマンドが送られ、MCBリモートからIOTスタンバイコマンド、デクレア

IPSリモートとスタンバイコントロール部102との間では、IPSリモートからイニシャライズフリーハンドエリア、アンサーコマンド、リムーヴエリアアンサーコマンド、カラー情報コマンドが送られ、スタンバイコントロール部102からはカラー検出ポイントコマンド、イニシャライズフリーハンドエリアコマンド、リムーヴエリアコマンドが送られる。

IPSリモートとセットアップコントロール部103との間では、IPSリモートからIPSレディコマンド、ドキュメント情報コマンドが送られ、セットアップコントロール部103スキャン情報コマンド、基本コピーモードコマンド、エディットモードコマンド、M/Cストップコマンドが送られる。

II Tリモートとスタンバイコントロール部102との間では、II Tリモートからプレスキャンが終了したことを知らせるII Tレディコマンドが送られ、スタンバイコントロール部102からサンプルスキャンスタートコマンド、イニシャ

MCBフォールトコマンドが送られる。

MCBリモートとサイクルコントロール部104との間では、サイクルコントロール部104からストップジョブコマンドが送られ、MCBリモートからはMADEコマンド、レディフォアネクストジョブコマンド、ジョブデリゲードコマンド、IOTスタンバイコマンドが送られる。

MCBリモートとフォールトコントロール部106との間では、フォールトコントロール部106からデクレアシステムフォールトコマンド、システムシャットダウン完了コマンドが送られ、MCBリモートからデクレアMCBフォールトコマンド、システムシャットダウンコマンドが送られる。

II Tリモートとコミュニケーションコントロール部107との間では、II Tリモートからスキャンレディ信号、イメージエリア信号が送られる。

次に各モジュール間のインターフェースについて説明する。

システムメイン１００から各モジュール（１０１～１０７）に対して受信リモートＮＯ．及び受信データが送られて各モジュールがそれぞれのリモートとのデータ授受を行う。一方、各モジュール（１０１～１０７）からシステムメイン１００に対しては何も送られない。

イニシャライズコントロール部１０１は、イニシャライズ処理が終了するとフォルトコントロール部１０６、スタンバイコントロール部１０２に対し、それぞれシステムステート（スタンバイ）を通知する。

コミュニケーションコントロール部１０７は、イニシャライズコントロール部１０１、スタンバイコントロール部１０２、セットアップコントロール部１０３、コピーサイクルコントロール部１０４、フォルトコントロール部１０６に対し、それぞれ通信可否情報を通知する。

スタンバイコントロール部１０２は、スタートキーが押されるとセットアップコントロール部１０３に対してシステムステート（プログレス）を

通知する。

セットアップコントロール部１０３は、セットアップが終了するとコピーサイクルコントロール部１０４に対してシステムステート（サイクル）を通知する。

### （Ⅲ）モニタ制御方式（本発明の要部）

本発明のモニタ制御方式は、処理時間を要するタスクは優先レベルの低い処理、１０ msec毎に行わなければならないモニタの処理は優先度の高い処理と位置付け、優先レベルの低い処理は、バックグラウンドとしてＣＰＵの空いている時間は常時実行し、１０ msec毎の割り込みがあると中断してモニタの通常処理を行うようにして割り込み毎のモニタの処理を支障なく行うと共に、処理時間の長いタスク処理も確実に実行できるようにしている。

また、本発明におけるモニタは、ＭＣＢ、ＵＩとの間ではＬＮＥＴによる通信、ＩＩＴやＩＰＳとの間では９６００ＢＰＳによる通信というように、他のシステムとの通信を行う場合に、通信速

度の異なる複数種類の通信方式を採用して分散ＣＰＵ方式の特徴を最大限に活かして効率的に機能させるようにしている。

第１４図は本発明のモニタ制御方式の原理を説明するための図である。

モニタは１０ msec毎の割り込みによりタイマ処理、入出力処理、送受信処理、ＡＰＰＳコール処理等通常の処理を行っている。そしてＩＰＳにおけるＴＲＣ変換テーブル作成処理のように、非常に処理時間を要するタスクをリザーブ（ＲＳＶ）タスクとして登録しておく。今、Ｔ。のタイミングでの割り込みでモニタの通常処理が行われ、ｔ。のタイミングでこの処理が終了すると、バックグラウンドの処理、即ちＲＳＶタスクの処理が実行される。そしてタイミングＴ。の１０ msec割り込みが入ると、バックグラウンド処理は中断され、モニタの通常処理が実行される。以後、割り込みによるモニタ通常処理が終了し、ＣＰＵの空いた時間を利用してＲＳＶタスクの処理が実行される。したがって、１０ msec割り込み毎のモニタの通

常処理は支障なく行われると共に、処理時間のかかるＲＳＶタスクはＣＰＵの空き時間を利用して行われ、ＣＰＵの稼働率を向上させることもできる。

### （Ⅲ－１）システムモニタのモジュール構成

第１５図はシステムモニタのモジュール構成を示す図である。

システムモニタは、計算機のＯＳに相当するもので、システムの利用者とハードウェアの間に位置してシステム資源の管理、スループットの向上、応答時間の短縮、使い易さ、汎用性、拡張性、情報の共有と保護、及びＲＡＳ（信頼性、利用可能度、保守性）の向上を狙いとしてシステムの管理に関する処理を行っており、本実施例においては、タスクのコントロール処理を行うタスクコントロールモジュールと通信処理を行う通信モジュールからなっており、通信モジュールはさらに３６００ＢＰＳ通信モジュール（９６００ＣＯＭ）とＬＮＥＴ通信モジュール（ＣＣＣ・ＣＯＭ）から構成されている。

タスクコントロールモジュールは、10 msec 割り込みによりタイマ処理、入出力処理、APPS コール処理等を優先度の高い処理として行い、同時に10 msec 割り込みによる処理が行われないCPUの空き時間に、パワーONからOFFまでの間、常時処理時間の長い処理を優先度の低い処理としてAPPSに実行させている。

9600 COMは、IIT、IPSとの間の通信処理を行っている。

CCC・COMは、MCB、UIとの間における通信処理を行っている。

#### (A) タスクコントロールモジュール

タスクコントロールモジュールは、パワーONにより起動されるモニタメイン処理部、10 msec 割り込み、キャリッジスキャン割り込み、NMI 割り込み、イメージエリア割り込み等の割り込み処理部、APPSに対するサービス処理部、割り込みに対する空処理部からなっている。そして、本発明のバックグラウンド処理は、モニタメイン処理部で行っている。

VMテスト等のコアテストを行う。またこれらのテスト処理と共に、APPSのエラー処理を行うRUN\$ERRORをコールする。この場合、パラメータを付してコールすることによりAPPSが何でコールされたか分かるようにしており、APPSは付されたパラメータをみて対応するエラー処理を行う。

コアテストが終了すると、モニタメイン処理部801はCCC・COM（コミュニケーションコントロールチップモジュール）に対してコンテンションバスのセルフテストを実行させるために、C\$SELF\$TEST1、C\$SELF\$TEST2をコールする。このテストは、所定のデータを送出して自ら受信し、受信したデータが送信したデータと同じであることを確認することにより行う。

これらの処理が終了すると、割り込みテスト処理部801-2が起動され、パワーON1回目のスキャンレディ、スキャンエンド、スキャンスタート、イメージ読み込み・開始・終わり等の外部

モニタメイン処理部801はパワーONにより起動され、パワーON初期化処理部801-1を起動し、周辺LSI初期化処理部801-5、割り込みコントローラ（8259）初期化処理部801-9、タイマコントローラ（8254）初期化処理部801-10をそれぞれ起動して各LSIの初期化処理を行うと共に、9600 bps通信モジュール（9600 COM）に対して9600 bps通信を行うための8251 INIT、COM\$INITをコールして初期設定処理を行わせ、同様に9600 COMに対してHDR\$SETをコールして予め決まったフォーマットのデータを設定しておくメモリの初期設定を行う。

RAMテスト処理部801-6、ROMテスト処理部801-7、NVMテスト処理部801-8はパワーON初期化処理部801-1からコールされて書き込んだデータがその通り読めるか否かのRAMテスト、前回の電源ON時の値と同じ値が読めるかどうかのROMテスト、電源を切る前のデータが正しく保存されているかどうかのN

割り込みに対して、回路が正しく動作するか否かのテストが行われる。

こうしてパワーオンステートで行うべきテストが終了すると、APPSのイニシャライズ処理を行うSS\$INIT\$SYS、SS\$INIT\$UIをコールする。ここで言うAPPSとはSYS及びUIを含んでおり、SYS及びUIのどちらもモニタからみるとAPPSということになり、これらを動かすための初期設定処理を行う。

イニシャライズステートになると、タイマテーブル初期化処理部801-3が起動され、タイマテーブルに、例えばFFFF等の所定の値がセットされる。

タイマセット後、モニタメインループ処理部801-4が起動され、10 msecを超える長い繰り返し処理を無限ループで行うことになる。無限ループは、10 msec割り込みによる通常処理で中断するものの、パワーOFFまでの間、常時裏で回っている処理である。

10 msec割り込み処理部802は割り込みコン

トローラ8259を介しての内部割り込みにより10 msec毎に起動され、タイマ処理部802-1、フィルタ処理部802-2、APPSコール処理部802-3、RSVステートチェンジ処理部802-4、RAMモニタ処理部802-5、出力処理部802-6、各種タイマチェック処理部802-7~802-9、各種タイマ消去処理部802-10~802-12の起動を行う。10 msec割り込み処理部802の活性化により、現在実行中の処理が一時中断してメインメモリの固定領域に退避し、タイマ処理、入出力管理、タスク管理等10 msec毎に行うべき処理が一巡して行われる。そして、10 msec毎の処理が終了すると、退避させておいた処理に復帰する。割り込みに伴う現在実行中の処理の退避および復帰はハードウェアにより自動的に行われる。

タイマ処理部802-1は各種タイマのチェック処理、消去処理を起動すると共に、タイマ値の登録、カウントダウン、タイムアウト処理等のタイマ処理を行う。

る処理が無効にならないようにしている。

以上は10 msec毎の内部割り込みから起動されるモジュールであるが、次にIITコントローラからの外部割り込みにより起動されるモジュールについて説明する。

キリッジが移動すると、IITからモニタに対して外部割り込みが入り、スキャンレディ割り込み処理部803、ノットスキャンレディ割り込み処理部804、イメージ割り込み処理、ノットイメージ割り込み処理も起動される。スキャンレディ割り込み処理部803、ノットスキャンレディ割り込み処理部804はキャリッジが動いた時に入ってくる割り込みで、各々立ち上がり、立ち下がりて起動される。

IR6割り込み処理部805、IR7割り込み処理部806は割り込みコントローラ8259が1つのLSIで8本の割り込みができるため、使っていなくても一応設定しておかないと、誤動作につながるので、割り込み処理のプログラムの器だけ作っておくための空の割り込み処理部である。

フィルタ処理部802-2は外部入力信号のノイズを拾わないように、同一レベルデータが所定時間継続したときにデータ取り込みを行うような処理を行う。

APPSコール処理部802-3はタイマ処理、フィルタ処理が終わった後、所定のAPPSをコールしてそちらに制御を渡す処理を行う。

RSVステートチェンジ処理部802-4は、バックグラウンドとして行われるべきRSVタスクの処理状態が停止中、実行中、実行状態、実行終了のいずれかに応じてステートチェンジを行う。

RAMモニタ処理部802-5は、RAMの内容を読みだして7セグメント表示するRAMモニタボードの表示処理を行うためのもので、ソフトウェアのデバッグ用に使用している。

出力処理部802-6は処理結果を出力するための処理を行う。

WDTリセット処理部802-13はプログラムの暴走を防ぐためのウォッチドッグタイマを10 msecの割り込み毎にリセットして、時間のかか

NMI割り込み処理部807はマスクできない割り込み(NON MASKABLE INTERRUPT)で、割り込みコントローラ8259からでなく、CPUに直接入ってくる割り込みである。この割り込みは緊急にCPUにリセットがかかってパワーが落ちたりする場合、その直前に入ってくる割り込みで、どういう状態でパワーOFFになるかということをNVMに記憶させておくための極く短時間の割り込みである。

イメージエリア割り込み処理部808は、イメージの読み込みはスキャンしたときに行うので精度を要するため、その処理をCCCからの割り込みの中で行っており、読み込みのトリガ指令を出すためのものであり、ノットイメージ割り込み処理部809は、イメージの読み込みが終了したことを指令するためのものである。

モニタはAPPSに対するサービス処理を行っており、APPSが10 msecに一回モニタからコールされたとき、その中でモニタのサービス処理、即ちスーパーバイザーコールを使ってAPPSが

モニタに対してタイマの登録、停止、バックグラウンドでの長い処理の実行を要求できる。

即ち、10 msec タイマ、100 msec タイマ、1000 msec タイマの各起動/停止処理部 810～812 は、APPS からのスーパーバイザーコールに対するタイマサービス処理を行っており、これを利用して APPS はタイマセット、停止を行うことができる。また、各起動/停止処理部 810～812 は、10 msec タイマ、100 msec タイマ、1000 msec タイマ初期化処理部及び消去処理部 802-7～802-12 と接続されていて、それぞれ初期化、設定値の消去が行われる。

また、スーパーバイザーコールにより RSV タスク登録処理部 813、RSV タスクステータスチェック処理部 814、RSV タスク実行要求処理部 815、RSV タスク処理部 816 がコールされ、それぞれバックグラウンドとして処理される RSV タスクの登録、RSV タスクステータスの確認、登録抹消、実行要求、タスクの停止等を行う。

9600 COM は、IIT、IPS との間で 9600 bps のシリアル通信を行い、通信割り込み処理、送受信割り込み処理、APPS との間のデータのやりとり等を行っている。

システムから IIT、IPS へのデータ送信開始は、精度を要求されるので割り込みで知らされている。即ち、タイマコントロール LSI 8254 から、ある一定時間がたつと IIT、IPS に対して通信をやりなさいという割り込みがかかってくるので、通信割り込み処理部 827 は、その制御下にあるモジュールを活性化してこの通信を開始するための一連の処理を行っている。

通信テスト処理部 827-1 はシリアル通信回線のテストを行い、通信リモート設定処理部 827-2 は 9600 bps のシリアル通信方式で通信するスレーブリモートを設定するための処理を行う。

回線エラー監視処理部 827-3 は通信エラーが何回か連続して通信が行える状態ではないと判断したときに回線エラーとする処理を行っている。

既に、第3図において説明したように、CCC は SYS 系、UI 系、MCB 系間の高速通信回線 LNET のプロトコルをサポートしており、コンテンションバス通信、エラー発生等の場合に割り込みを行うサービス機能を有しているが、現在のところこの割り込みサービス機能は使っていない。しかしながら高速通信回線 LNET を通してノイズ等が割り込み信号として入ってきた場合に、これに対応する処理部がないと割り込みがあったとしてシステムが暴走してしまう可能性がある。そこで、システムモニタは、このような CCC からの割り込みを処理する割り込み処理部 817～822 を有しており、ここに割り込み信号が入った場合には、そのままリターンさせる処理を行っている。

8086 内部割り込みから起動される割り込み処理部 823～826 も同様にノイズ等の割り込みがあった場合の空処理を行うための処理部である。

#### (B) 9600 BPS 通信モジュール

ここで、通信エラーとは一回の通信に対して発生するエラーのことで、3種類のものがある。

##### (1) 無応答エラー

リモートからの応答がない場合のエラーである。

##### (2) ステータスエラー

リモートから受信したデータが誤っているエラーで、①パリティエラー、②ストップビットが検出されないフレーミングエラー、③通信用バッファに送られてきたデータを取り込まないうちに次のデータがきてしまった場合のオーバーランエラーの3種類がある。

##### (3) フレームエラー

フレームの論理に誤りがある場合のエラーで、例えばデータが途中で切れてしまい正常なバイト数分だけ受信できなかったような場合である。

このような通信エラーが、例えば3回以上連続したか否かチェックし、連続していれば回線エラーとしている。

通信開始処理部 827-4 は、リモートとの間の通信を行うためのもので、送信開始時に通信テ

ストを行う場合には通信テストコマンドセット処理部 827-5 をコールしてテスト用コマンドをセットし、同じデータを送る場合には再送データセット処理部 827-6 をコールして再送データをセットし、APPS からデータを送ってくれという要求があった場合には APPS データセット処理部 827-7 をコールして APPS からの送信データをコマンドと共にフレームにセットし、また受信したフレームからコマンドとデータを切り出す処理を行い、リモートに対して送るデータがない場合には、ACK セット処理部 827-8 をコールして ACK を送るための処理を行う。そして、データを送る場合には必ず HEADER 通信処理部 827-9 をコールして送信相手リモート No をヘッダにセットして送り、また受信したフレームからヘッダを取り出して相手リモートを判断する処理を行っている。

LSI 8251 は送受信の割り込みを行っており、送信割り込み処理部 830 はシリアル通信用 LSI 8251 に送信データを書き込み、送信デ

ータを 9600 通信ラインにのせると、次のデータを 8251 に書いても良いという割り込みがかかってくるので、その割り込みの中で IIT、IPS へ送る次のデータを 8251 へ書き込む処理をしている。この割り込みは 1 バイト毎にかかってくるので、送るデータが 5 バイトデータであれば 5 回の割り込みがかかってくる。

受信割り込み処理部 831 は送信処理とは逆に IIT あるいは IPS から 1 バイト 8251 へデータが送信されてくると、1 バイトデータを受信したという割り込みがかかってくるので、これを 1 バイト毎に受け取り、また次の 1 バイトが揃うと割り込みが入ってきて、順次 1 バイト単位でデータを受信する処理を行っている。

この受信割り込み処理部 831 の下でデータ受信開始設定処理部 831-1、データ受信処理部 831-2、受信エラーリカバリー処理部 831-3、受信リモート設定処理部 831-4、データ受信終了処理部 831-5、SELF テスト結果受信処理部 831-6 がそれぞれ機能しており、

その中で受信割り込み処理部 831 はデータが揃った場合に間違ったデータがきていないかどうか、あるいは IIT からのデータか IPS からのデータか等のチェックをしている。SELF テスト結果受信処理部 831-6 はパワー ON イニシャライズの中に通信テストという機能があるが、各 APPS を動かす前にシステムとの間で通信可能かどうかのテストを行うと共に、IIT、IPS のパワー ON セルフテストの結果を受信する処理を行っている。

LSI 8251 イニシャライズ処理部 832、送信イニシャライズ処理部 833、ヘッダーフォーマット設定処理部 834 は、それぞれパワー ON イニシャライズの中でタスクコントロールモジュールから呼ばれるイニシャライズ処理を行っている。

回線エラー検知処理部 835 は、タスクコントロールモジュールの 10 msec 割り込みの中で呼ばれ、モニタから APPS に対して回線エラーの発生を知らせる処理を行っている。

モニタの通信についてのみ考えると前述の回線エラー監視モジュールさえあれば、モニタは回線エラーかどうか把握して通信を止めたりすることができる。しかし、例えば IIT との間で回線エラーが発生し、このことを APPS が知らないと、APPS は IIT との通信は生きていると思って IIT に対してキャリッジスキャンの要求を出し、これに対してスキャン終了という返事が来ないので何時までもスキャンの要求を出したままの待機状態となり、メッセージ表示も「コピーしています」のままになってしまうという不都合が発生する。そこで、タスクモジュールにより 10 msec 毎に回線エラー検知処理部 835 をコールして、回線エラー監視モジュール 827-2 で生成した情報を APPS のエラー起動処理を行う RUN\$ERROR をコールして APPS に対して知らせるようにしている。

また、回線エラー検知処理部は、CCC 側の通信エラーが起きたときにいつまでもエラーにしておくと、MCB、UI との通信が何もできなくな

ってしまうので、エラーということが分かったらリカバリさせるために10 msec毎にCCC・COMのCCC\$ERCV\$CHKをコールしている。

送信処理部836は、送信チェックSVC836-1、送信SVC836-2はそれぞれタスクコントロールモジュールの10 msec割り込みの中で呼ばれる処理である。

IIT送信キューイング処理部837、IPS送信キューイング処理部838は、それぞれAPPSからのスーパーバイザーコールで呼ばれる処理で、バッファを持っていて100 msecに1回程度のデータ送信を行っている。

メイン送信キューイング処理部839、スベア送信キューイング処理部840はそれぞれスベア用であり、リモートが増えたときのために用意している。

IIT受信データ取り込み処理部841、IPS受信データ取り込み処理部842はそれぞれIIT、IPSからの受け取ったデータがあるかどうかというAPPSからのスーパーバイザーコールに対

ルフテスト処理、受信処理、送信処理を行っている。

パワーON時にメインモニタからコールされるタスクコントロールモジュールからCCCセルフテスト1モジュール850とCCCセルフテスト2モジュール851とがコールされる。

CCC・COMはセルフテスト、HALT、NORMALの3つのステータスを有しており、それぞれのステータスに対応したテストを行っている。

CCCセルフテスト1モジュール850は、CCCの初期設定処理を行うモジュールで、通信するときにCCCが正常に動作するか否かをテストする。まず、CCCステータスチェンジテスト1モジュール850-1はステータスをセルフステートからHALTに変えることを指示し、指示通りにステータスが変化するか否かのテストを行う。

次に、HALTステータスにおいて、ターンアラウンドテストモジュール850-2～850-4によりCCCチップに対してFIFOコマンドを書き込み、コマンド通りに動いてその通りの結

して受信データ提供のサービス処理を行っている。メイン受信データ取り込み処理部843、スベア受信データ取り込み処理部844は、受信データ取り込みのスベア用として設けられており、送信の場合と同様にリモートが増えたときのための用意として設けている。

#### (C) LNET通信モジュール

LNET高速通信は、第16図に示すように、システム、UI、MCBとの間で187.5 KbpsでCCCを介して行われている。CCCチップはソフトを有しており、SYSモニタからデータをCCCチップのバッファに書き込む場合には1パケット単位でデータを送っており、そのデータが送信し終わるとバッファ空の割り込みがかかってくる。また1パケット単位で受信した場合にも割り込みがかかってくる。さらに、通信エラーが発生した場合にもエラーの割り込みがかかってくるので、LNET通信に関する限りは、CCC・COMからみた場合にはモニタはAPPSのような扱いとなる。そして、CCC・COMは、セ

果が返ってくるか否かをテストする。

次に、CCCステータスチェンジテスト1モジュール850-1はステータスをHALTからNORMALに変え、その通りステータスが変化するか否かのテストを行う。そして、モジュール850-11、850-12によりそれぞれHALT、NORMALの初期設定を行う。

なお、FIFOコマンド書き込みチェックモジュール850-9、データ変換処理部850-10、エラー処理部850-6、ウェイト処理部850-7、リード変換処理部850-8はそれぞれCCCチップの仕様が他チップのものと異なっているため、共通処理を可能にするための変換を行っているインタフェースである。また、1 msecウェイト処理部850-7で直接ウォッチドッグタイマのポートにアクセスしてリセットしているが、これはチェックを開始してからそれが確定するまでに時間がかかるので、ウォッチドッグタイマのリセットを定期的に行わないとプログラムの暴走と認識されてしまうからである。



セルフテスト2モジュール851は、LNETのターンアラウンドテストを行っており、通信がうまくいくか否かをみている。このうち、モジュール851-1はイネーブル状態で、モジュール851-3はディセーブル状態で単にコマンドを実行させて、結果がOKかNGかを判断している。またモジュール851-2は、送信バッファ0、1のテストを行うためのデータ送信処理部851-4、851-5及び受信バッファ0、1のテストを行うためのデータ受信チェック処理部851-6、851-7を起動し、LNETがUI、MCBとの回線の他に自分の所に戻ってくる回線があるのを利用して、自分で送ったものを自分で受けて、送ったものがその通りに受け取れるか否かをテストしている。このテストにより、バッファから送信し、回線を通して受信する処理ができるか否かのテストを行うことができる。

なお、セルフテスト2においてもインタフェース850-6～850-8を共用しており、またデータ変換処理部851-8を共通処理のための

受信割り込み処理部853は、バッファ0またはバッファ1へのデータ受信があった場合に、受信バッファ処理部853-1または853-2をコールしてバッファ受信キューイング処理部853-3-853-5をコールしてデータの取り込みを行っている。この場合、MCBは、システムに対してリクエストセルフテストリザルト、バックグラウンドポーリング(BGP)という特別のコマンドを定期的に送ってきて、システム側のLNET高速通信が生きているか否か聞いてくること、UIに対して通信データ量が多いためキューイング処理部を2つ設けている。

9600BPSの送信処理がタスクコントロールモジュールから10msec毎にコールされた結果9600BPSから、あるいはLNET割り込みからCCCの送信処理を行うSEND\$CCCモジュール854がコールされる。

この送信処理部854は、各バッファ854-1～854-6にどのデータをどのように書き込むかの処理を行っており、第17図により詳細に

インタフェースとしている。

LNETの割り込みには①エラーが発生したとき、②送信バッファが空になったとき、③データ受信があったときの3種類がある。

LNETの割り込みがあると、各モジュールは自分に対するものであるか否か判断してその処理を行う。

エラー割り込み処理部852-1は、例えばデータ再送回数オーバーの場合、データが送れなかったということで、バッファ0または1のエラー処理部852-2または852-3を起動し、またエラー復旧処理部852-4を起動する。なお、リード変換処理部852-5は前述の共通処理のためのインタフェースである。

使われていない割り込みがかかってきたとき、エンドインタラプトコマンドをCCCチップのレジスタに書き込んで返してやる必要がある。そのためにタスクコントロールモジュールよりインタフェースとしてのデータ変換処理部852-6をコールして変換処理を実行させている。

説明する。なお、モジュール852-6は前述した共通処理のためのインタフェースである。

第17図(a)はLNET通信を説明するための図、第17図(b)は送信リクエストと送信バッファ0、1の空き具合による送信パターンを示す図である。

第17図(a)に示すように、CCCは送信バッファと受信バッファをそれぞれ2つ有しており、1パケット単位でデータが書き込めるようにしている。なお、送信バッファ0はMCB用、送信バッファ1はUI用として使用している。バッファのデータが空になると、モニタに対して割り込みがかかってきて次のデータが送信可能であることを知らせる。そして、1つのバッファのデータをMCB、或いはUIに対して送信している間、他のバッファへの書き込みを行うことにより送信処理が効率的に行えるようにしている。このデータの送信に対してMCB、或いはUIから返すものがない場合にはACKコマンドがCCCに対して返される。受信についても同様で、1パケット単

位でデータが送られてくると、そのことが割り込みで知らされ、2つのバッファから交互にデータの取り込みが行えるようにしている。

第17図(b)において、EはEMPTY、FはFULL、B0はバッファ0、B1はバッファ1、○は送信要求あり、×は送信要求無し、鉤状矢印はバッファの選択をそれぞれ意味している。送信方法は、リクエストセルフテストリザルトとMCBからのバックグラウンドポーリング(BGP)に対する応答は最優先とし、MCB、UIに対する送信要求については、要求が単独で有るときには空いているどちらのバッファも使用し、MCB、UIの両方の要求があるときはMCB用バッファ0のみ空のときはMCBへ送信し、UI用バッファ1のみ空のときはUIへ送信する。

なお、送信処理は10msecからもLNETからも割り込みによりコールされるようにしている。これは、パワーONのときや最後のデータを送ってしまった場合には、データが何もないのでバッファFULLからバッファEMPTYという変化

が生ぜず、そのためLNET割り込みがかからず、データの送信処理ができない。このときは10msecの割り込みがトリガーとなって送信処理をコールしている。また、10msec割り込みだけであれば10msecに一回しか送信できない。しかしLNET割り込みがあることにより、データを送ってバッファが空になれば直ぐ割り込みがかかるので、データが沢山あっても連続的に送信することができる。なお、10msec割り込みとLNET割り込みが同時に起きた場合にはLNETに対してはマスクして競合が生じないようにしている。

エラー復旧チェック処理855は、エラー処理においてエラーの復旧が行われた否かのチェックをしている。

BGP送信データ変換処理部856、MCB、UIへの送信、受信処理を行うモジュール857～860は、APPSに対するサービス処理であるスーパーバイザーコールによる送受信処理を行っている。また、モジュール859-1は前述した共通処理のためのインタフェースである。

#### (Ⅲ-2) タスクコントロール状態遷移

第18図はタスクコントロールの状態遷移を示す図である。

本発明におけるモニタのタスクコントロール部の処理は第18図(a)に示すバックグラウンド処理ループと、第18図(b)に示す10msec割り込み処理ループとで実行される。

第18図(a)において、バックグラウンドループ処理はパワーON時から行われ、第15図(a)のモニタメイン処理部801により起動される各モジュール801-1～801-10により周辺LSIの初期化、RAM、NVMテスト、APPS初期化等を行うパワーON初期設定処理と、同様にモニタメイン処理部801により起動されるモニタメインループ処理部801-4により、初期設定後に実行要求のある処理時間の長いRSVタスクの実行処理を繰返し行い、TRC変換テーブル作成のように処理時間が10msecを越えてしまう長い時間を要する処理を行う。

また、第18図(b)において、10msec割り

込み処理ループは10msec割り込み処理部802により起動される各モジュールにより実行され、10msec毎のリアルタイム割り込み処理内で、タイマ処理部802-1、フィルタ処理部802-2、APPSコール処理部802-3、出力処理部802-6によりそれぞれ10msec、100msec、1000msec計時処理、タイマ処理、入力処理、APPSコール処理、出力処理等の10msec処理、及びWDTリセット処理部802-13によりウォッチドッグタイマのリセットを行う100msec処理を実行している。

モニタはこのような2つの流れで処理を行い、RSVタスクの実行は、モニタメインループ処理部801-4における無限ループでパワーOFFまで実行し、割り込みがあると実行制御権を10msec処理部に渡してモニタの通常処理が行われる。

#### (Ⅲ-3) タスクテーブル

第19図はRSVタスクテーブルを示す図である。

本発明においては、バックグラウンドで処理す

る10 msecを超える長い処理を行う場合には、その処理をRSVタスクとして予めモニタRAMのタスクテーブルに登録しておく必要がある。このタスクテーブルはRSV0からRSV15までの16個用意され、モニタRAMに登録できるタスクの個数は最大16個までとしている。

第19図(a)(c)はRSVタスクステートテーブルを示しており、第19図(a)はモニタが作成するRSVタスクステートテーブル、第19図(c)はAPPS参照用に設定されるRSVタスクステートテーブルを示し、それぞれモニタRAMに設定される。

タスクは停止(0)、実行待ち(1)、実行中(2)、実行終了(3)のステートを有しており、現在どのステートにいるかは0~3の値をステートテーブルに設定することにより分かるようにして、各RSVタスク毎にその値がモニタRAMに設定される。従って、このステートテーブルを参照して0~3の何れの値であるかを見ることにより、RSVタスクがどういう状態にいるかを判断

することができる。

なお、RSVタスクステートテーブルを2つ用意しているのは、後述するように、完全に処理が終了したのか、中断したのか分かるようにタスクのステートはモニタが10 msec間だけ実行終了の状態を保持している。これはAPPSは10 msecに1回しか起動されないで、10 msec間保持しておかないと認識できないからである。この10 msec間ステートを保持するためにテーブルを2つ設けている。即ち、モニタはステートが変化する毎に1つのテーブルの内容を書き換え、他のテーブルは10 msec毎に書き換えている。なお、APPSは第19図(c)のテーブルを参照してRSVタスクの状態を判断しているが、モニタの方はAPPS参照用のタスクステートテーブルは参照していない。

第19図(b)はループカウンタテーブルを示しており、ループカウンタは各タスクに対して16個用意され、後述するようにそれぞれ最大256まで処理ループ回数をカウントしている。

第19図(d)はRSVタスクアドレステーブルである。RSVタスクはバックグラウンドループ処理の初期設定においてモニタRAMに設定され、その登録RSVタスク数は登録時にカウントされて幾つのRSVタスクが登録されたか分かるようになっている。

なお、第19図(a)(c)(d)のテーブルはモニタRAMに設定され、第19図(b)のループカウンタテーブルはモニタとAPPS共有RAMに設定される。これはRSVタスクの登録、実行、停止等がAPPSからのスーパーバイザーコールでモニタの行うサービス処理で行われ、そのためRSVタスクアドレステーブル、RSVタスクステートテーブルはモニタRAMに設定される。一方、ループカウンタはAPPSの処理(バックグラウンド処理)の中で生成されるカウント値、即ち何回の処理ループが実行されたかをモニタが管理しておく必要があり、そのためモニタが管理するテーブルに書き込むので、モニタとAPPS共有RAMに設定している。

#### (Ⅲ-4) TRCの設定及び計算処理フロー

第20図はIPSのTRC設定処理のフローを示す図である。

前述したように各RSVタスクには16個のループカウンタが用意されており、ここではループカウンタ0を使用し、原稿の種類等に応じて標準カーブを修正するために、通常4点の濃度、コントラスト、カラーバランスの設定処理を行う。さらに要求があればネガポジ反転処理を行い、1回の処理が終了するとループカウンタ0の内容をプラス1して、以後順次この処理を繰返し、ループカウンタ0の内容が256階調数を越えると処理が終了する。

第21図は中間点補間計算処理、標準カーブ計算処理、反転計算処理を行うフローを示す図である。

中間点補完計算処理は、設定点が4点であるのでその中間を埋めるための直線近似処理である。ここではこの処理をループカウンタ1を使用し、階調数が256であるので256点について中間

点補完処理を行う。

標準カーブ処理は、例えばループカウンタ2を用い、中間点補完処理で求めた256点についての出力値を標準カーブにより変換して標準カーブの修正を行い、TRC変換テーブルを作成する処理を行う。

反転計算処理は、例えばループカウンタ3を用い、得られたTRC変換テーブル（カーブ）の傾きを90度変える処理で、例えば濃度の濃い所を逆に薄くしたいような場合に行う。

このように、IPSのTRC計算処理は、濃度、コントラスト、カラーバランス等の設定処理、256階調を出すための中間点補完処理、標準カーブ修正処理、反転処理256階調で行うので膨大な処理量となり長い処理時間を要することになる。

#### (Ⅲ-5) RSVタスクの状態遷移

制御のしやすさという観点から、RSVタスクの状態は停止、実行待ち、実行中、終了の4に分けて、RSVタスクの管理を行っている。

第23図は10msec割り込み処理、バックグラウンド処理、他の割り込み処理の関係を示す図である。

10msec割り込みによりモニタの通常処理が行われる。割り込みの瞬時のみディスエイブルとなって直ちにイネーブルとなり、他の割り込みを受け付け可能としている。そしてRSVタスク実行要求(RSV EXCT)がでると、10msec割り込みによるモニタの通常処理が終了すると、バックグラウンド処理がCPUの空き時間に行われ、次に10msec割り込みがあるとモニタの通常処理が実行される。また、他の優先する割り込みがあると10msec割り込みによるモニタの通常処理、バックグラウンド処理が中断してその割り込み処理が優先して行われる。

RSV EXCTが消去されると、RSVタスク処理は終了するが、バックグラウンド処理は継続して行われ、CPU空き時には空ループが行われることになる。

第23図から分かるように、本発明において、

第22図はRSVタスクの遷移条件を示す図である。

パワーONからパワーオフまで継続的に実行されるバックグラウンドにおけるRSVタスク処理は、パワーON時には「停止0」の状態であり、APPSがRSVタスクの実行要求をすると、タスク登録カウンタをプラス1して「実行待ち1」の状態に遷移する。「実行待ち1」の状態ではAPPSがRSVタスクの実行を中止する場合はタスク登録カウンタをマイナス1して「停止0」に戻る。「実行待ち1」の状態ではモニタがRSVタスクの実行を開始すると、「実行中2」の状態に遷移する。この状態でRSVタスクの実行が行われ、モニタがRSVタスク実行を終了する時はタスク登録カウンタをマイナス1して「実行終了3」の状態に遷移し、モニタが10msec間だけ実行終了を保持して停止にする。これは完全に処理が終了したのか、中断したのかが分かるようにするためである。

#### (Ⅲ-6) 割り込み処理とバックグラウンド処理

バックグラウンド処理が1番低いレベルの処理、10msec割り込みによるモニタの通常処理がその上位の処理、他の優先割り込みが最優先の処理というように階層的になっている。

#### (Ⅲ-7) バックグラウンドループと割り込みループ処理フロー

第24図はメインループ（バックグラウンドループ）処理と、10msecループ処理の処理フローを示す図である。

パワーONによりバックグラウンド処理ループが駆動し、初期設定が行われ、APPS初期設定によりタスクアドレスがタスクアドレステーブルに設定される（ステップ1001）。そしてディセーブルにし、16個について順番に実行要求があるかないかみていく。そのため最初に、RSV0タスクがあるかないかをRSVタスク登録カウンタの値を見て判断し、タスクがあれば次にRSV0タスクの実行要求があるかないかをRSV0タスクステートが「実行待ち」にあるかどうかで判断し（ステップ1002～1004）、RSV

0タスクの実行要求があればRSV0タスクステートを実行中とする(ステップ1005)。次に割り込みをイネーブルとし(ステップ1006)、RSVを実行する(ステップ1007)。なお、RSV0タスクのステートを実行中にするのにディセーブルとして行っているのは、割り込みの方でもRSV0タスクステートを動かすことがあり、それと重ならない状態で処理をするためである。こうして、前述したループによる処理を行う。

そして処理が終了すると、ステートを変える必要があるので、前述と同様に割り込みと処理が重ならないようにディセーブルとし(ステップ1008)、RSV0のステートが「実行中」かどうか見て、RSV0ステートが「実行中」であればRSV0タスクステートを「終了」とする(ステップ1009、1010)。これはRSV0タスク実行中でもAPPSの方からその実行を止めてくれという指示もできるので、止まっているものを「終了」とする必要がないので「実行中」か否

Vタスクの停止要求をだしてループカウンタにFFFF0Hをセットする。もちろん10msec割り込みが入った時点でバックグラウンド処理から10msec割り込み処理に制御は渡されているので、RSVタスクの処理は中断してステップ1008のディセーブルの状態にいる。また、RSVタスクの実行要求もAPPSコール処理の中で行われ、RSVタスク実行要求があれば登録カウンタの値をカウンタアップし、同時にRSVタスクステートテーブルを「実行待ち」の状態とする。

次に送信キューイングのエリアに書き込まれたデータを送信ホームセット処理により送信する(ステップ1127)。そしてRSVステートチェンジ処理によりAPPS参照用RSVタスクステートテーブルのステートの書き換えを行う(ステップ1128)。そしてRAMモニタ処理、ポーリングチェック、エラー検知処理、出力処理等を行い(ステップ1129~1132)、100msec経過したか否か判断し、経過している場合はウォッチドッグタイマのリセットを行い(ステッ

かを判断している。こうして、RSV登録カウンタの値をマイナス1し、割り込みをイネーブルとして(ステップ1012)、以後RSV0タスクと同様にRSVタスク1~15について、要求のあったタスクについての処理を行う。

一方10msec割り込みループでは10msecのリアルタイム割り込みがあると、まず10msec割り込みのマスクをセットし、自分自身の割り込みを禁止する(ステップ1101)。これは10msec割り込み処理が長引いた場合に次の自身の10msec割り込みが入ると、戻れなくなってしまうのでそれを防止するためである。そしてイネーブルとして他の割り込みを受付可能とし(ステップ1102)、次にステップ1103~1123でタイマの計時処理を行う。

計時処理が終了すると10msec、100msec、1000msecのタイマ処理を行い、次にノイズをデータとして読み込むのを防止するフィルタ入力処理を行う(ステップ1125)。そしてAPPSコール処理を行い(ステップ1126)、RS

ブ1133、1134)、割り込みをディセーブルとし、かつ10msec割り込みマスクをリセットし(ステップ1135、1136)、10msec割り込みをイネーブルとしてリターンする。

#### (Ⅲ-8) タイミングチャート

第25図は正常系実行待ちにおける中断、実行中の中断の場合のRSVタスク処理を示すタイミングチャートである。

第25図(a)は正常系の場合を示し、まずパワーONでループカウンタの値は0となっており、RSVタスクステートは停止の状態となっている。ここで、APPS10msec処理の中からモニターバイザーコールによりRSVタスクの実行要求があるとループカウンタをクリアし、RSVタスクステートを実行待ちの状態とする。ここで、モニタバックグラウンドモジュールがRSVタスクをコールしてループ処理を実行させるとRSVタスクステートは「実行中」となり、ループカウンタはループする毎にインクリメントする。そしてRSVタスクからモニタバックグラウンドに対

して制御が返されると、RSVタスクステートを「終了」とし、モニタは10 msec間終了保持した後、ステートを停止にする。こうして1つのRSVタスクの実行が終了する。カウンタの内容が最大値になれば、そこでRSVタスクは終了し、モニタバックグラウンドはRSVタスクステートを「実行中」から「終了」に変える。

第25図(b)は実行待ちの中断を示すタイミングチャートを示す図である。

パワーONでRSVタスクステートが停止の状態、APPS10 msec処理の中からモニタSVCに対して実行要求がなされ、ループカウンタ0クリア、RSVタスクステートを実行待ちとする。そしてRSVタスクステートが待ちの時に、APPS10 msec処理により中断要求が出されると、モニタSVCはRSVタスクステートを待ちの状態から停止とし、ループカウンタの値は0FFF0Hとし、処理は中断する。

第25図(c)は実行中の中断のタイミングチャートを示す図である。

パワーONでRSVタスクステート停止の状態、APPS10 msec処理からモニタSVCに対してRSVタスク実行要求が出されると、モニタSVCはRSVタスクステートを待ちにし、ループカウンタを0クリアする。ここで、モニタバックグラウンドからRSVタスクがコールされ、ループ処理が実行され、同時にRSVタスクステートが実行中に切換えられる。そしてループ処理毎にループカウンタがインクリメントし、処理が行われる。そしてAPPS10 msec処理の中で中断要求が出されると、モニタSVCはこれを受けてRSV中断要求を出し、ループカウンタの内容を0FFF0Hとし、RSVタスクからモニタバックグラウンドに対してタスクがリターンし、RSVタスクステートを実行中から終了に切換え、モニタは終了を10 msec間保持した後、ステートを停止とする。

〔発明の効果〕

以上のように本発明によれば、処理時間を要するタスクは優先レベルの低い処理、10 msec毎に

行われなければならないモニタの処理は優先度の高い処理と位置付け、優先レベルの低い処理は、バックグラウンドとしてCPUの空いている時間のみ実行し、10 msec毎の割り込みがあると中断してモニタの通常処理を行うようにして、割り込み毎のモニタの処理を支障なく行くと共に、処理時間の長いタスク処理も確実に実行でき、CPUの稼働率を向上することもできる。さらに、通信処理をコンテンツンバスによる高速通信とポーリング方式による通信を使用することにより、分散CPU方式の特徴を活かして、効率のよい通信処理を行うことができる。

#### 4. 図面の簡単な説明

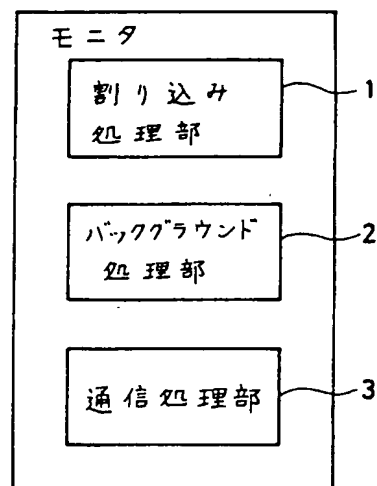
第1図は本発明のモニタ制御方式の構成を示す図、第2図は本発明が適用されるカラー複写機の全体構成の1例を示す図、第3図はハードウェアアーキテクチャを示す図、第4図はソフトウェアアーキテクチャを示す図、第5図はコピーレイヤを示す図、第6図はステート分割を示す図、第7図はパワーオンステートからスタンバイステ

ートまでのシーケンスを説明する図、第8図はプログレスステートのシーケンスを説明する図、第9図はダイアグノスティックの概念を説明する図、第10図はシステムと他のリモートとの関係を示す図、第11図はシステムのモジュール構成を示す図、第12図はジョブモードの作成を説明する図、第13図はシステムと各リモートとのデータフロー、およびシステム内モジュール間データフローを示す図、第14図は本発明のモニタ制御方式の原理を説明するための図、第15図はシステムモニタのモジュール構成を示す図、第16図、第17図はLNET通信を説明するための図、第18図はタスクコントロールの状態遷移を示す図、第19図はRSVタスクテーブルを示す図、第20図はIPSのTRC設定処理のフローを示す図、第21図は中間点補間計算処理、標準カーブ計算処理、反転計算処理を行うフローを示す図、第22図は状態遷移を示す図、第23図は10 msec割り込み処理、バックグラウンド処理、他の割り込み処理の関係を示す図、第24図はメインループ

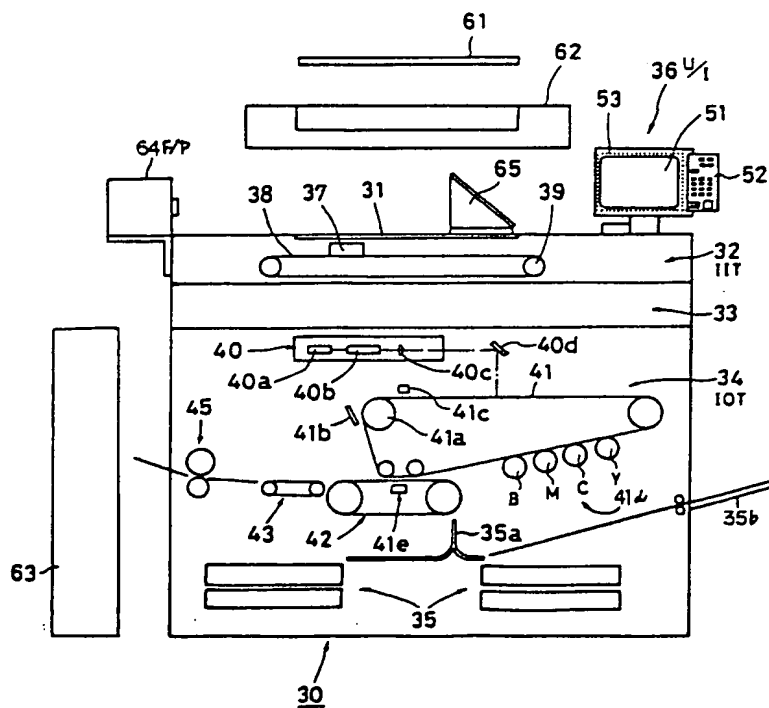
処理と、10 msecループ処理の処理フローを示す図、第25図は正常系実行待ちにおける中断、実行中の中断の場合のRSVタスク処理を示すタイミングチャートを示す図、第26図は処理時間の長いタスクが発生した場合のモニタによるタイム処理における不具合の発生を説明するための図である。

出 願 人 富士ゼロックス株式会社  
代理人 弁理士 蛭 川 昌 信 (外5名)

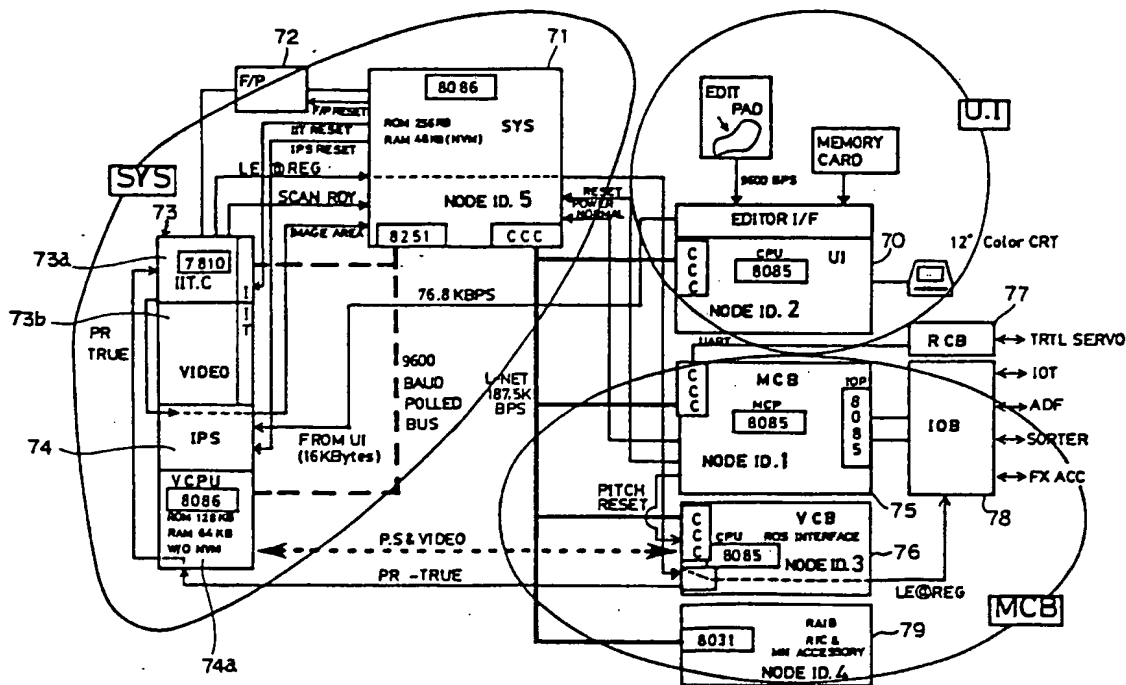
## 第 1 図



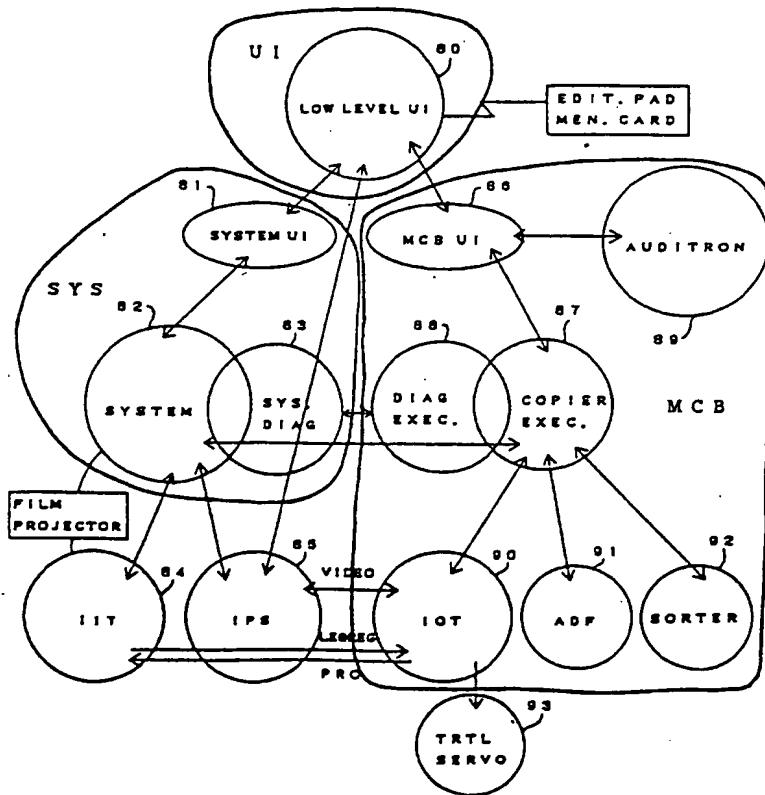
## 第 2 図



第 3 図

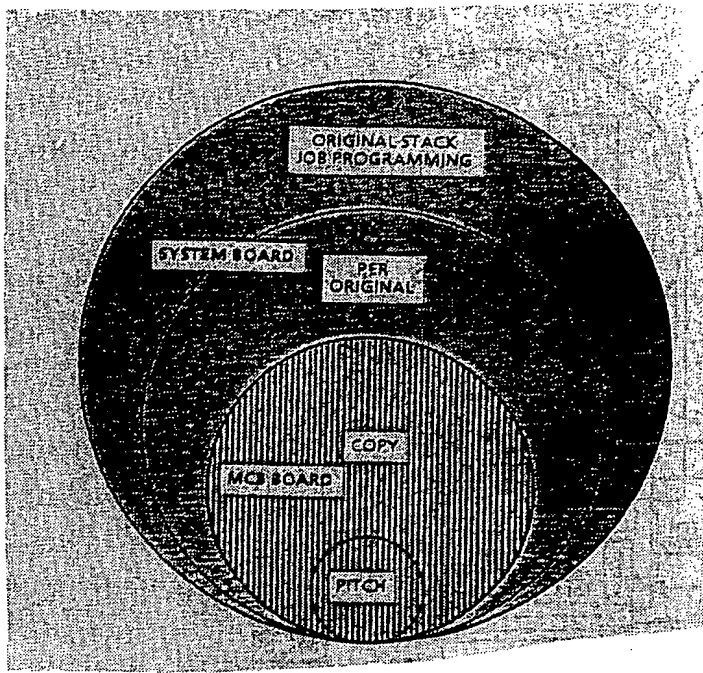


第 4 図

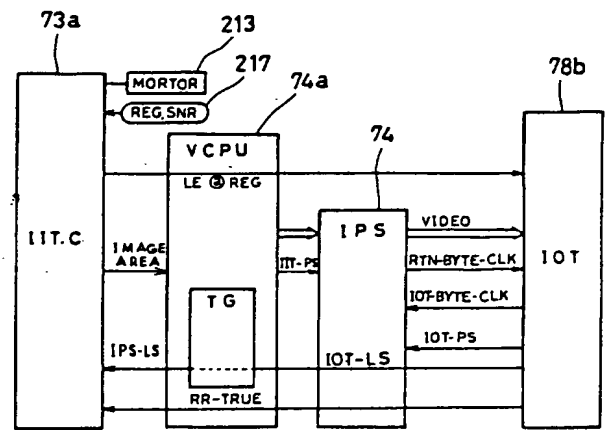




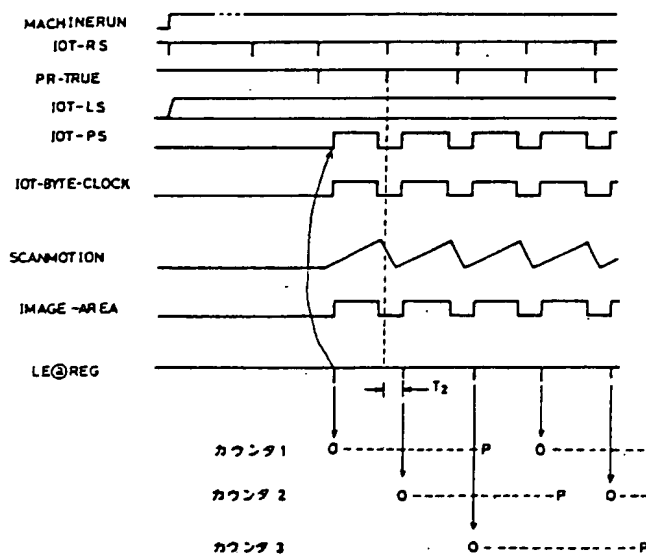
第 5 図 (a)



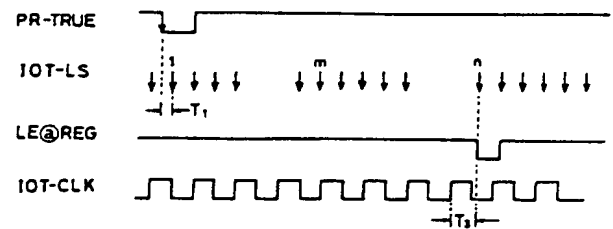
第 5 図 (b)



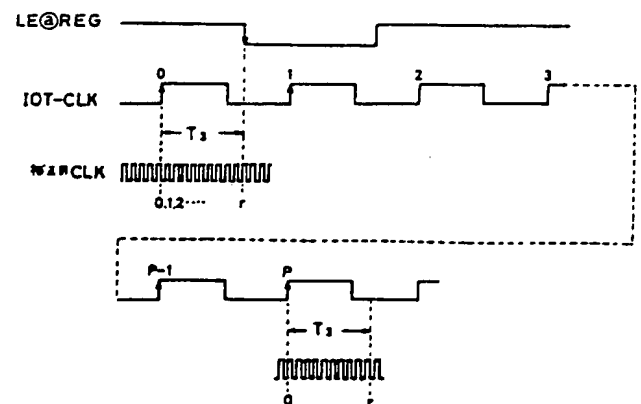
第 5 図 (c)



第 5 図 (d)

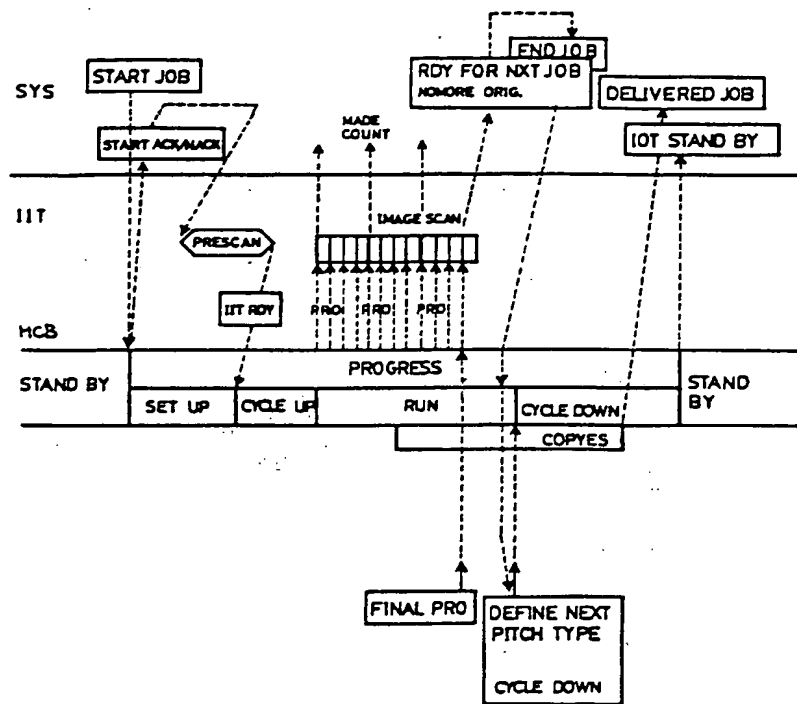


第 5 図 (e)

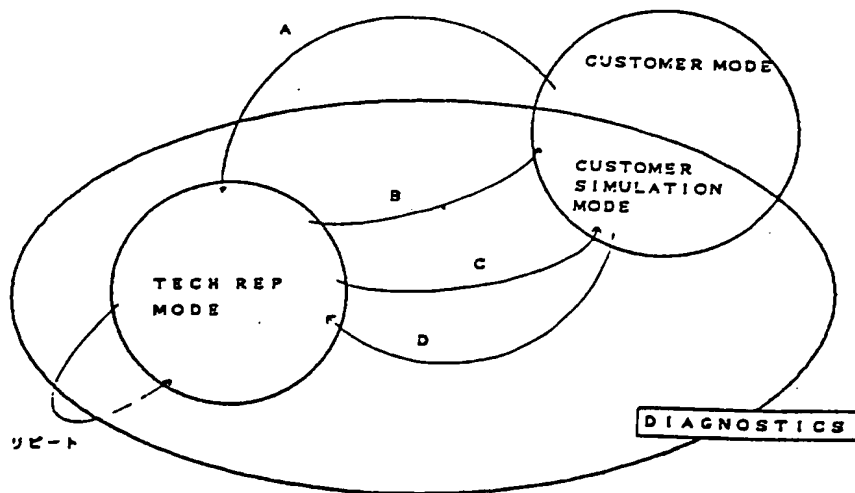




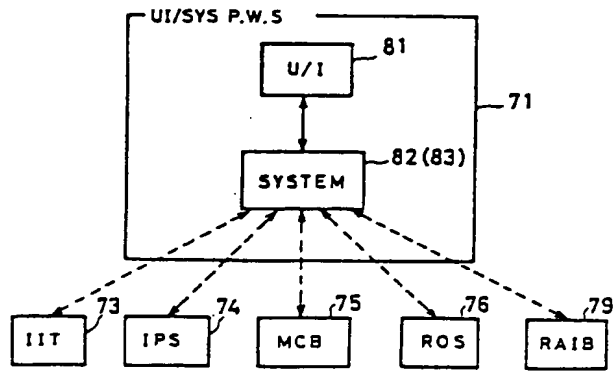
# 第 8 図



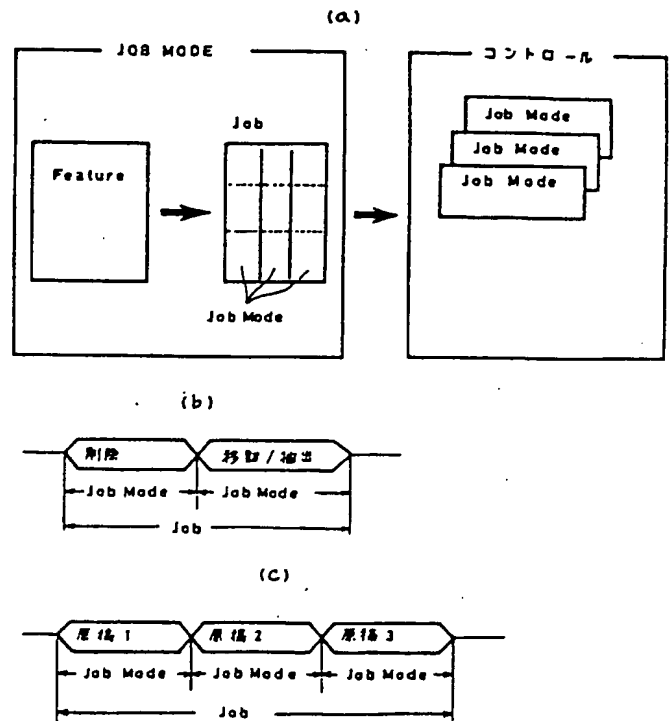
# 第 9 図



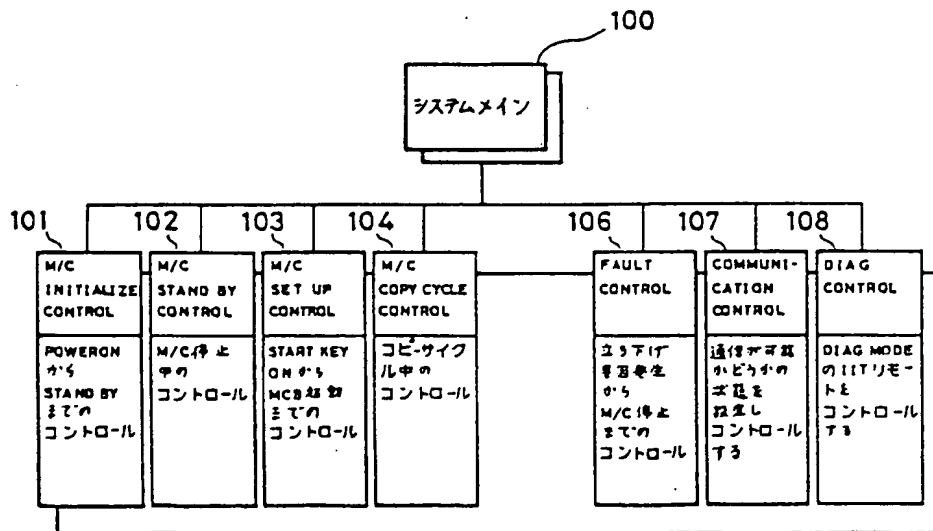
第10図



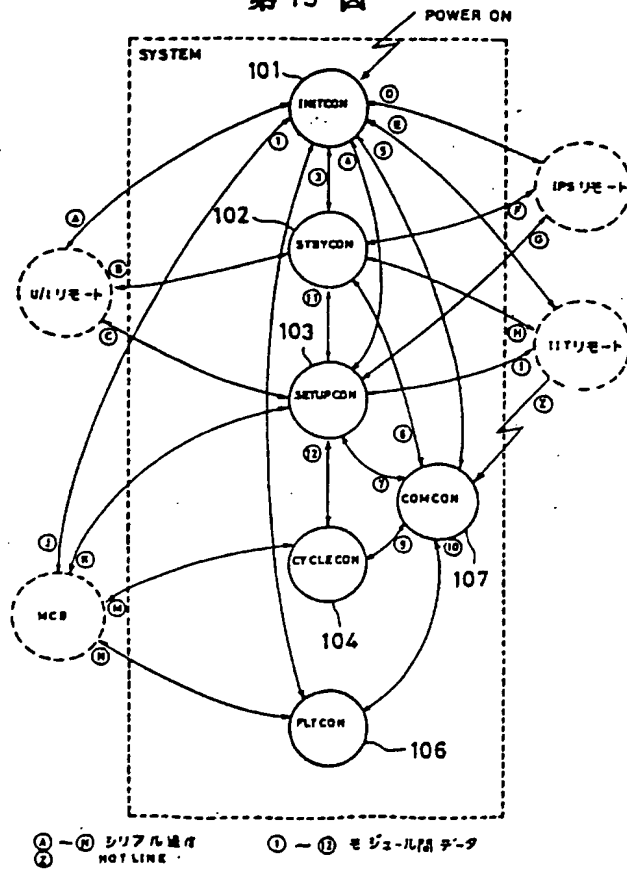
第12図



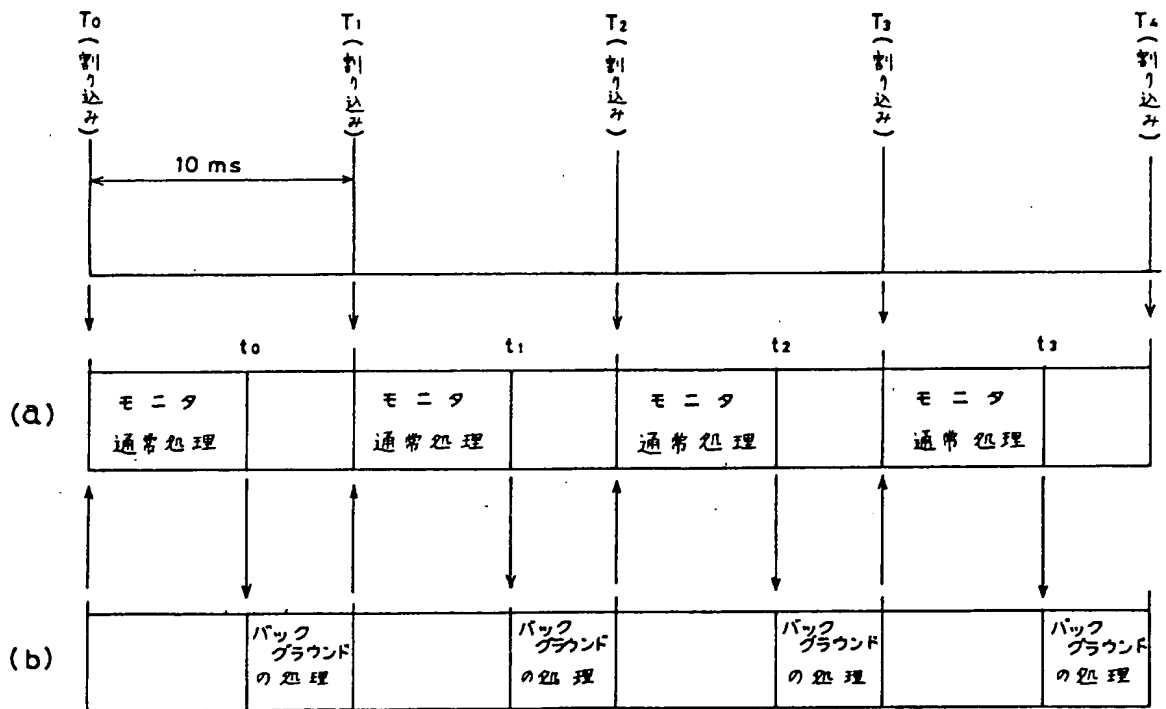
第11図



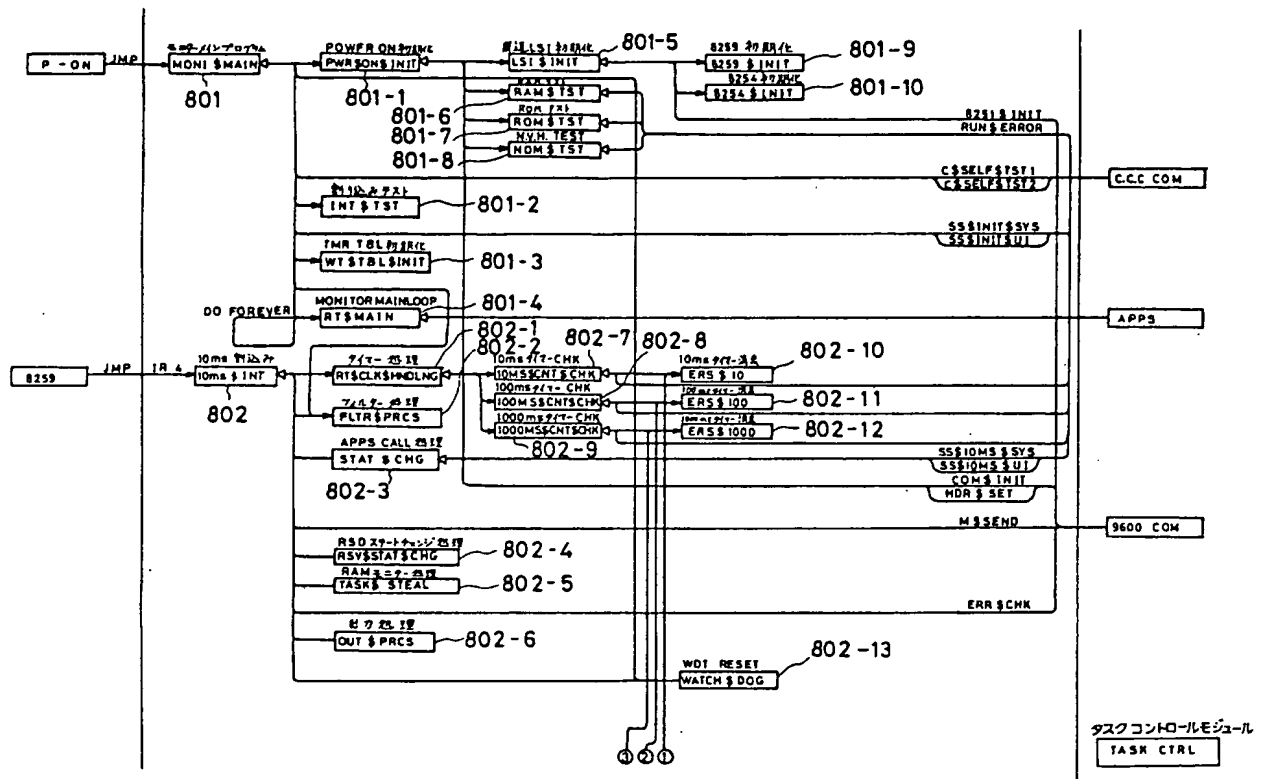
第13図



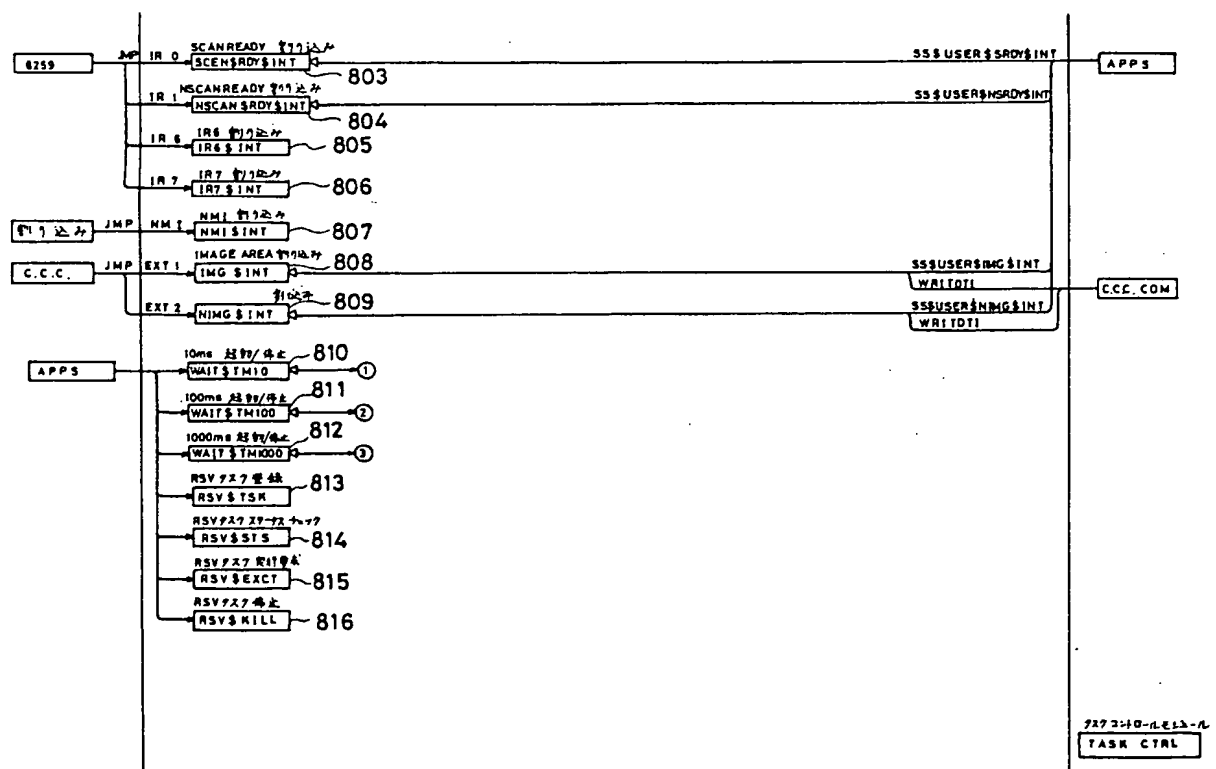
第14図



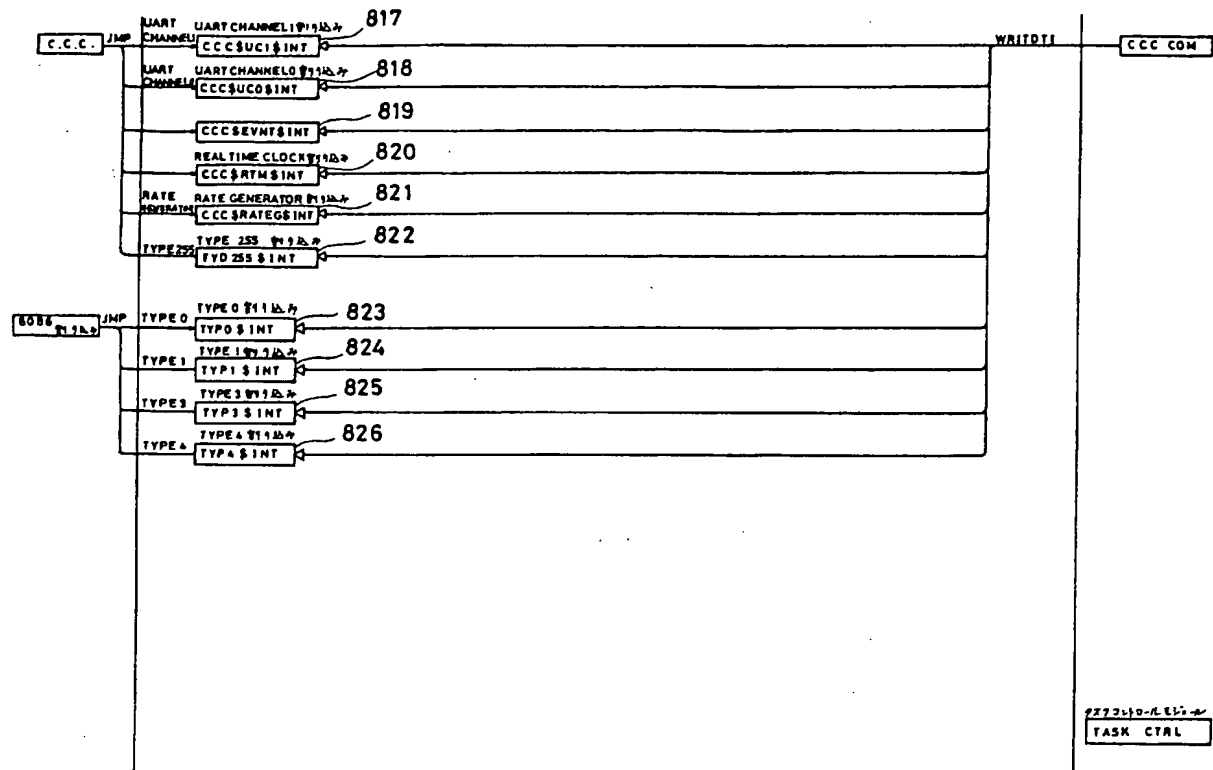
第15 図 (a)



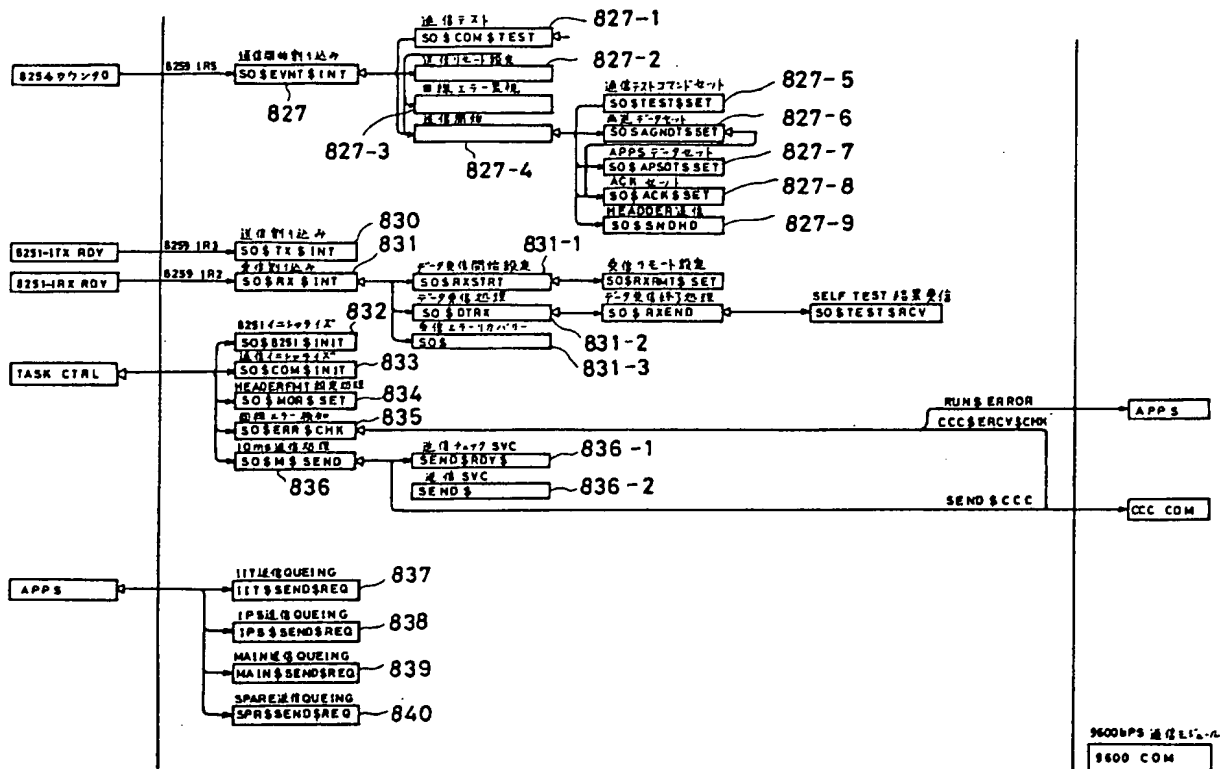
第15 図 (b)



第15 図 (c).



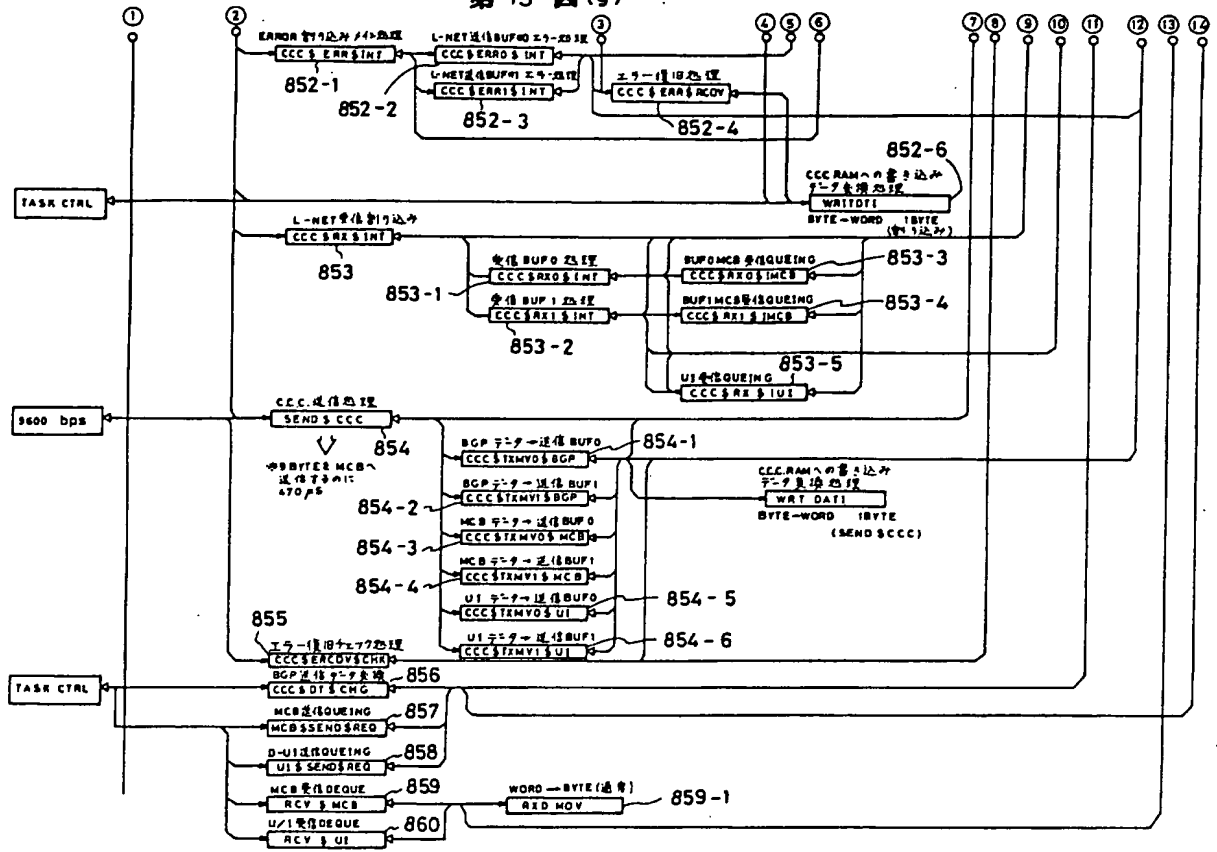
第15 図 (d)



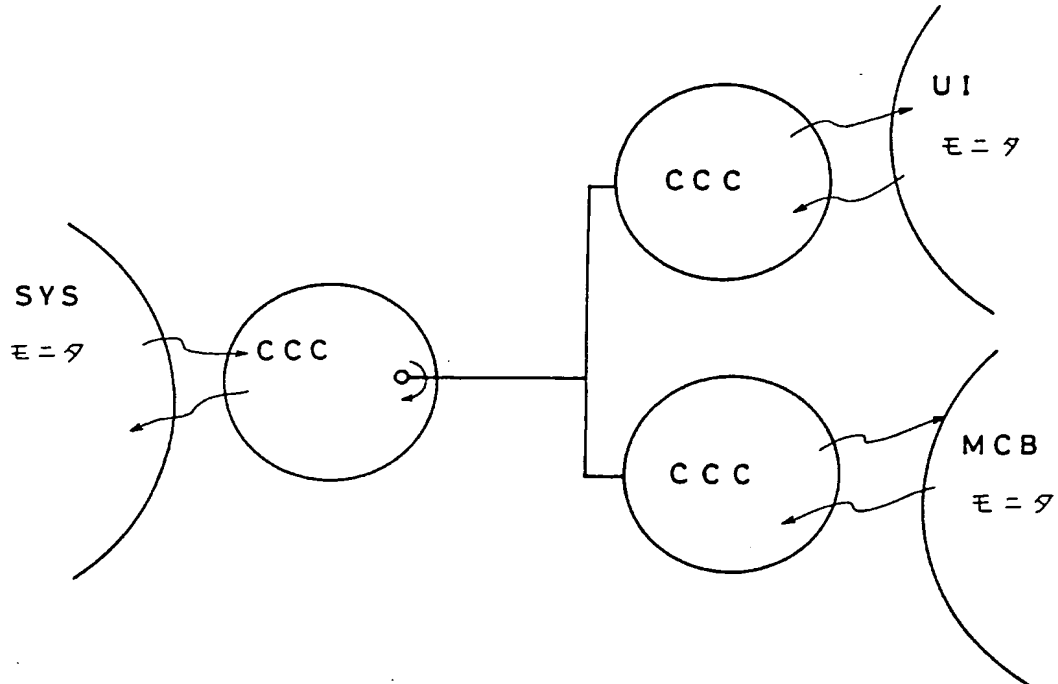




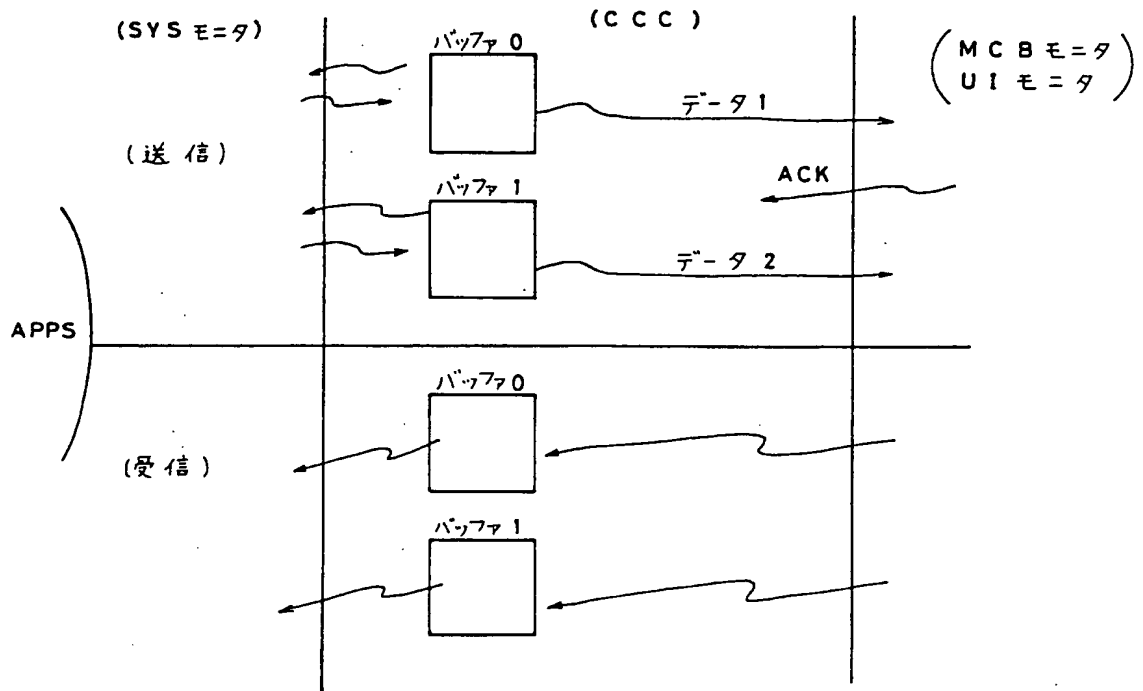
第 15 図 (g)



第 16 図



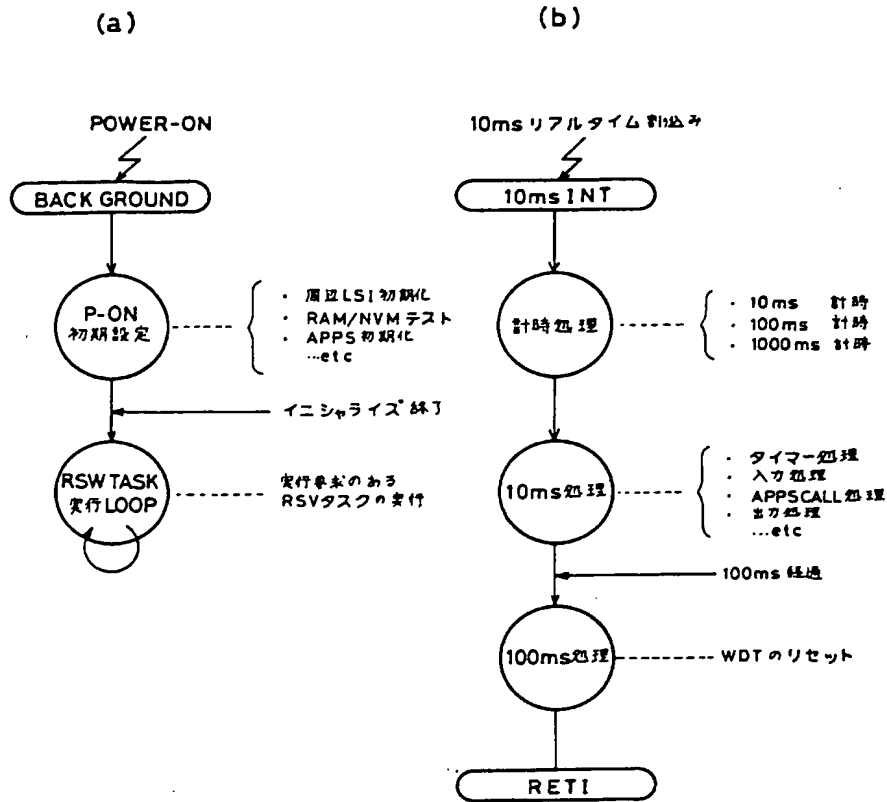
第 17 図 (a)



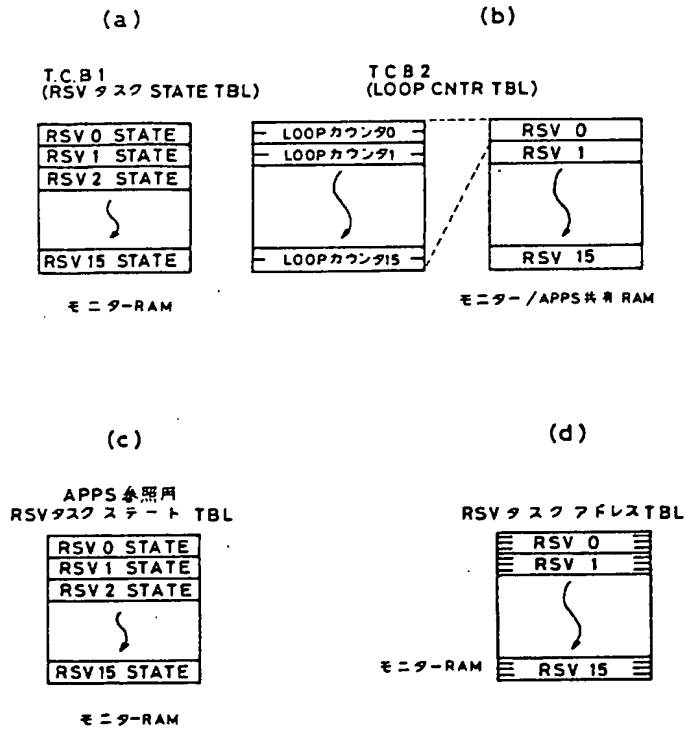
第 17 図 (b)

<div>&lt;送信リクエスト&gt;</div> <div>&lt;送信BUF空の割合&gt;</div>			CASE A																								
			0			1			2			3			4			5			6			7			
			BGP	MCB	UI	BGP	MCB	UI	BGP	MCB	UI	BGP	MCB	UI	BGP	MCB	UI	BGP	MCB	UI	BGP	MCB	UI	BGP	MCB	UI	
			×	×	×	○	×	×	×	○	×	×	×	○	○	×	○	×	○	×	○	○	○	○	○		
CASE B	3	B0	E	×			<div>↓</div>			<div>↓</div>			<div>↓</div>			<div>↓</div>			<div>↓</div>			<div>↓</div>			<div>↓</div>		
		B1	E	×			<div>↓</div>			<div>↓</div>			<div>↓</div>			<div>↓</div>			<div>↓</div>			<div>↓</div>					
	1	B0	E	×			<div>↓</div>			<div>↓</div>			<div>↓</div>			<div>↓</div>			<div>↓</div>			<div>↓</div>					
		B1	F	×			<div>↓</div>			<div>↓</div>			<div>↓</div>			<div>↓</div>			<div>↓</div>			<div>↓</div>					
	2	B0	F	×			<div>↓</div>			<div>↓</div>			<div>↓</div>			<div>↓</div>			<div>↓</div>			<div>↓</div>					
		B1	E	×			<div>↓</div>			<div>↓</div>			<div>↓</div>			<div>↓</div>			<div>↓</div>			<div>↓</div>					
	0	B0	F	×			<div>／</div>			<div>／</div>			<div>／</div>			<div>／</div>			<div>／</div>			<div>／</div>					
		B1	F	×			<div>／</div>			<div>／</div>			<div>／</div>			<div>／</div>			<div>／</div>			<div>／</div>					

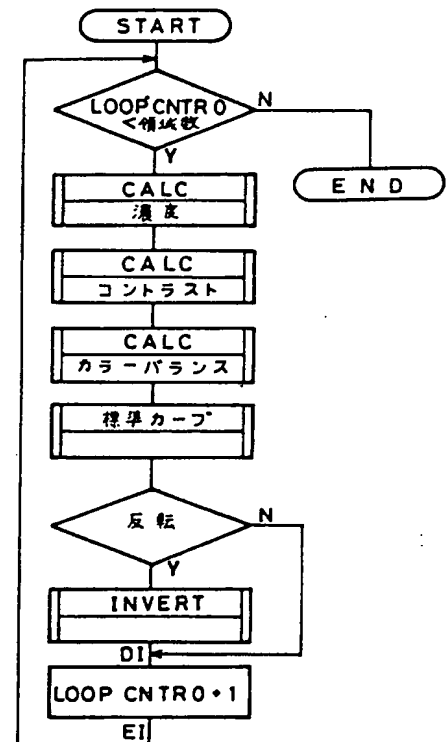
# 第18 図



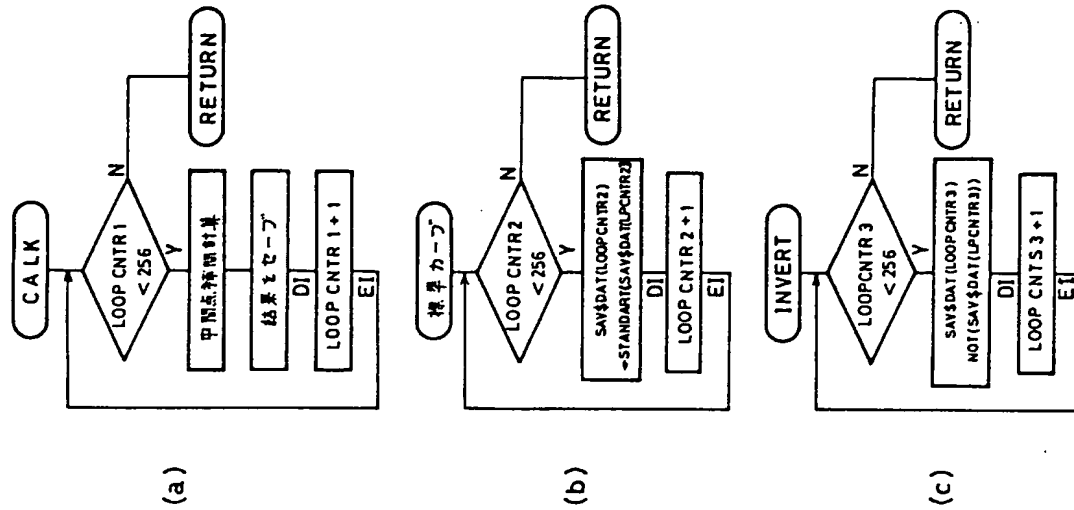
## 第19 図



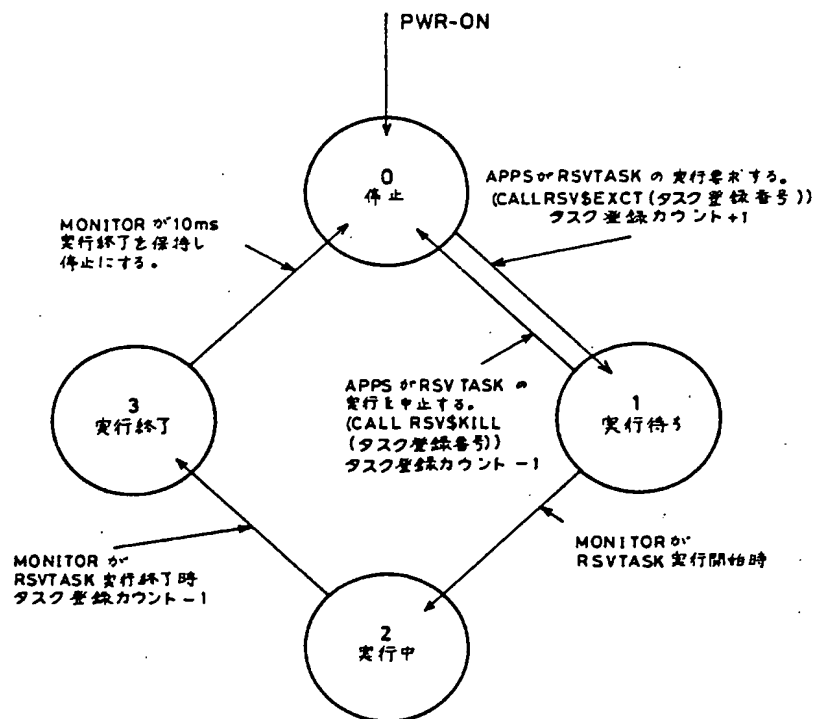
## 第20 図

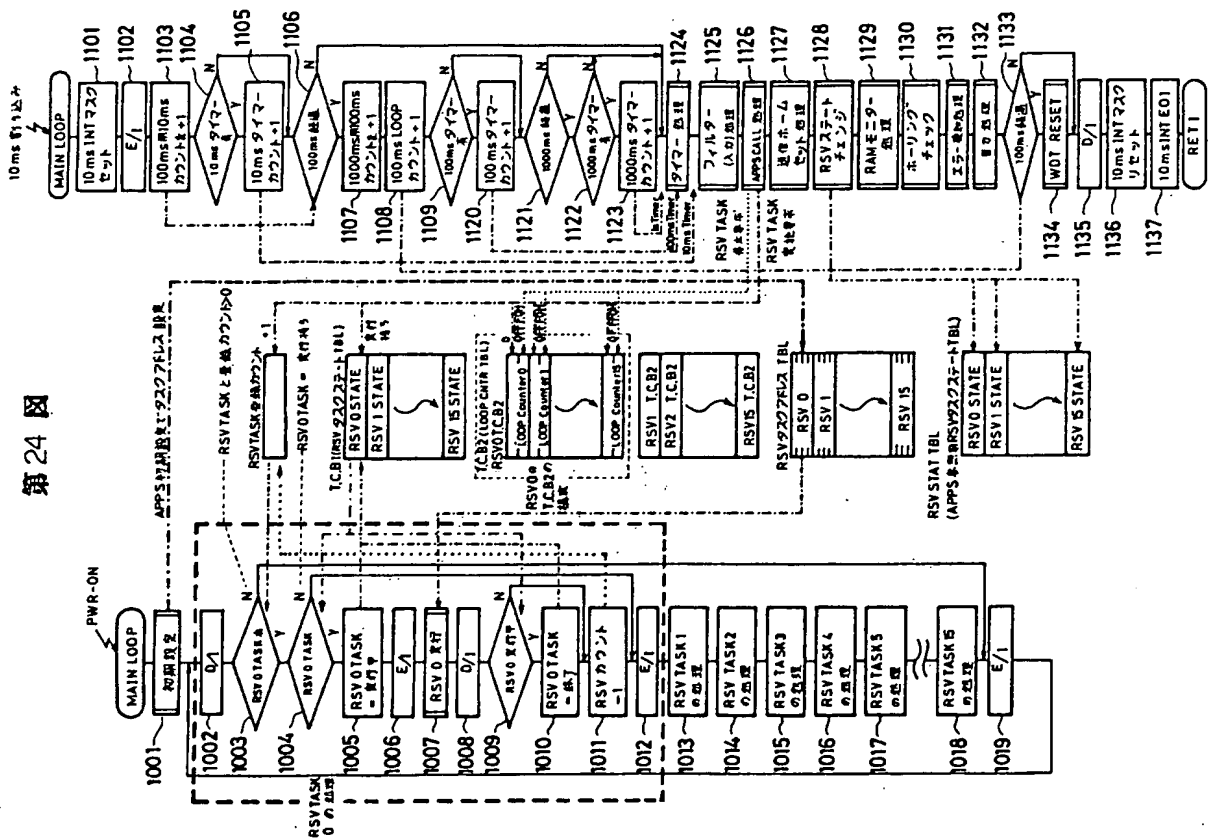
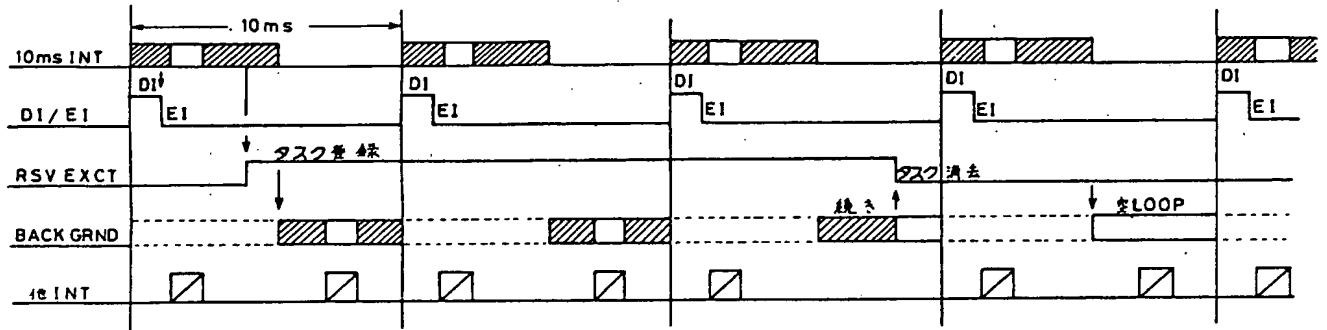


第21 図



第22 図

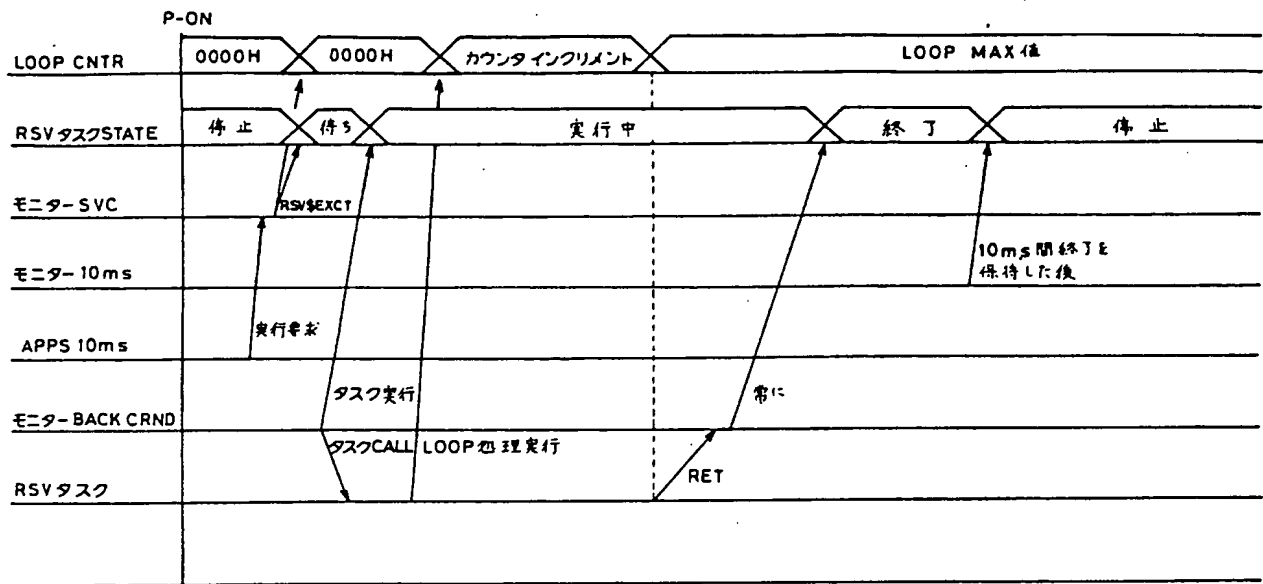




第24 図

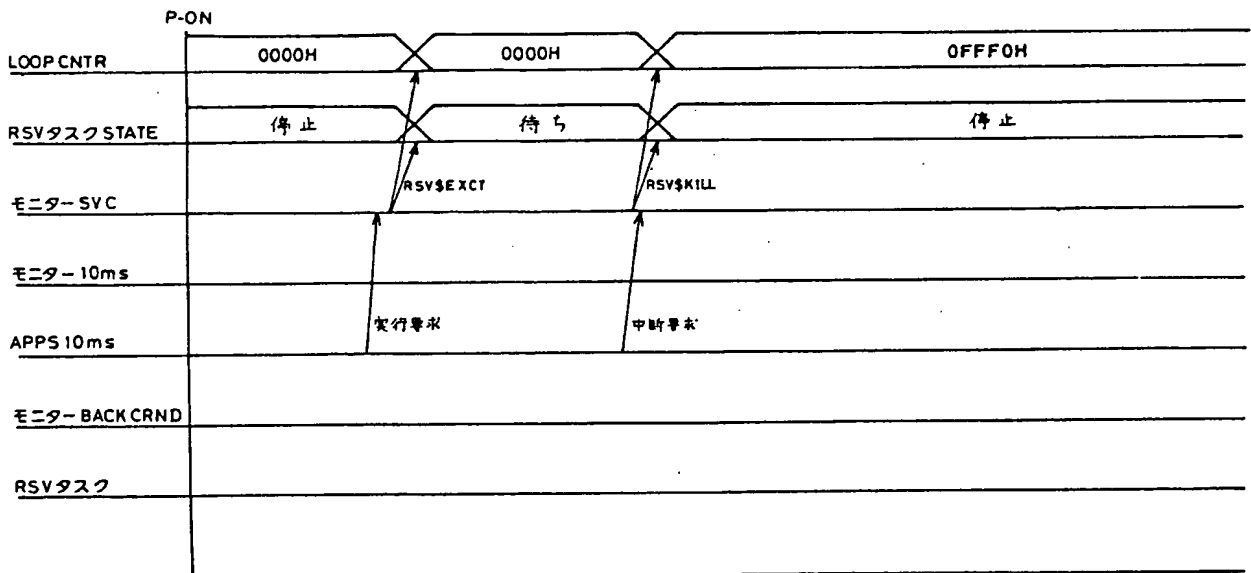
# 第25 図 (a)

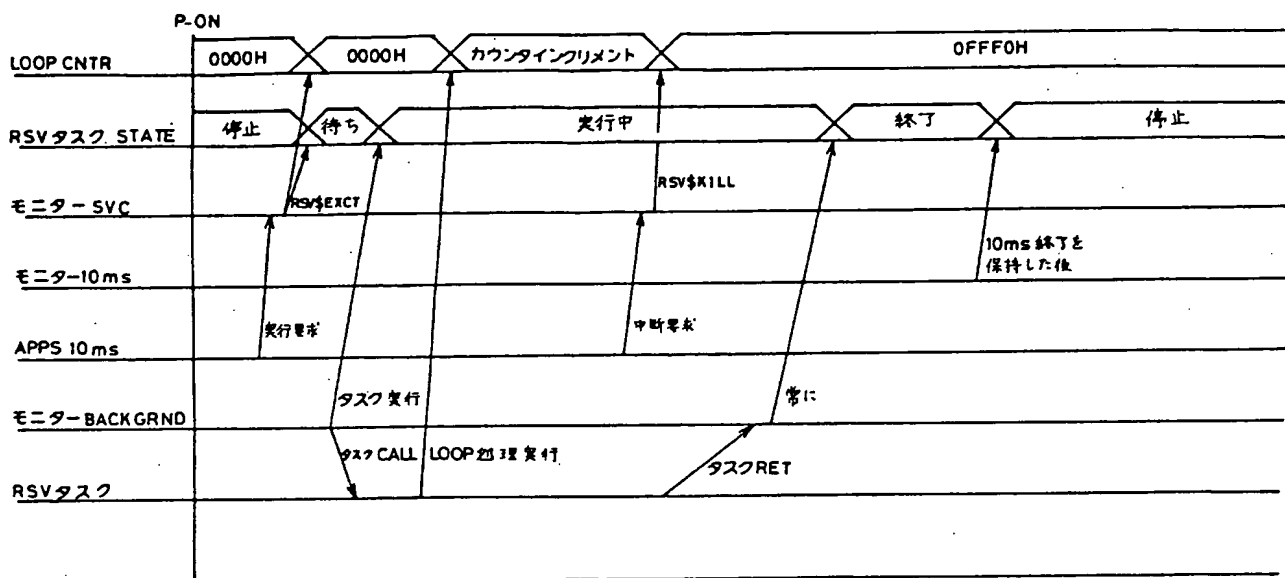
(正常系)



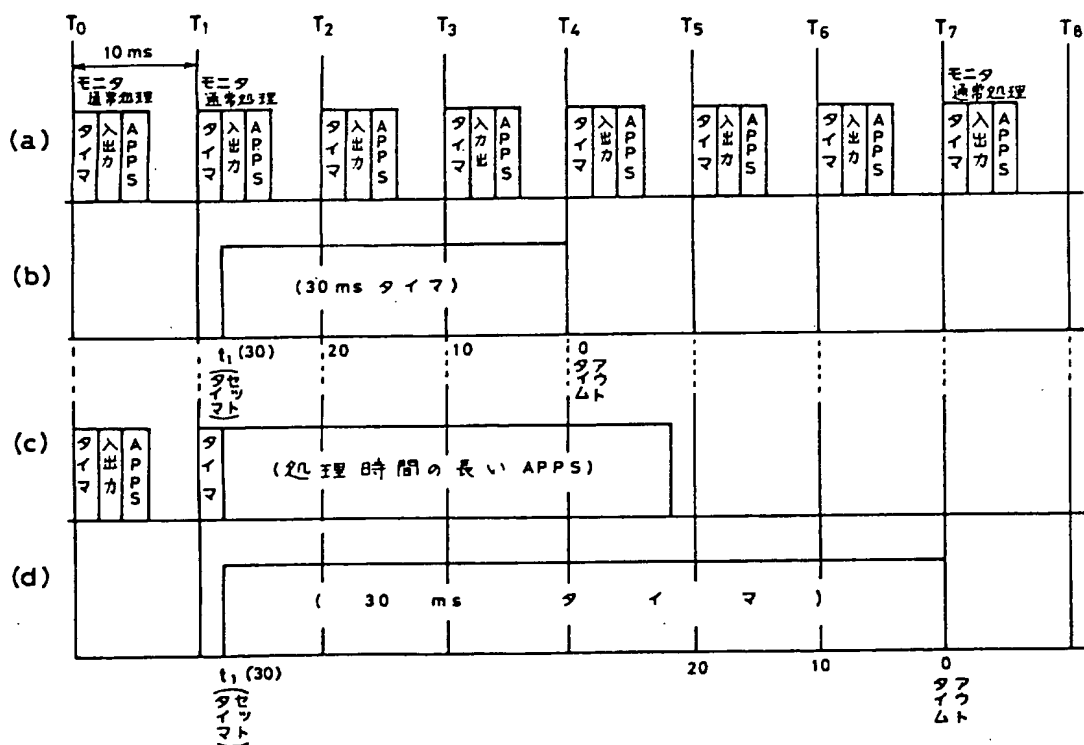
# 第25 図 (b)

(実行待ち中断)





第26 図



**This Page is Inserted by IFW Indexing and Scanning  
Operations and is not part of the Official Record**

**BEST AVAILABLE IMAGES**

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ BLACK BORDERS
- ☐ IMAGE CUT OFF AT TOP, BOTTOM OR SIDES
- ☒ FADED TEXT OR DRAWING
- ☒ BLURRED OR ILLEGIBLE TEXT OR DRAWING
- ☐ SKEWED/SLANTED IMAGES
- ☐ COLOR OR BLACK AND WHITE PHOTOGRAPHS
- ☐ GRAY SCALE DOCUMENTS
- ☒ LINES OR MARKS ON ORIGINAL DOCUMENT
- ☒ REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY
- ☐ OTHER: \_\_\_\_\_

**IMAGES ARE BEST AVAILABLE COPY.**

**As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.**